

PATENT ABSTRACTS OF JAPAN

(11)Publication number : **09-128276**

(43)Date of publication of application : **16.05.1997**

(51)Int.Cl.

G06F 12/00

(21)Application number : **08-191845**

(71)Applicant : **TOSHIBA CORP**

(22)Date of filing : **22.07.1996**

(72)Inventor : **IMAI TORU**

YOSHIDA HIDEKI

(30)Priority

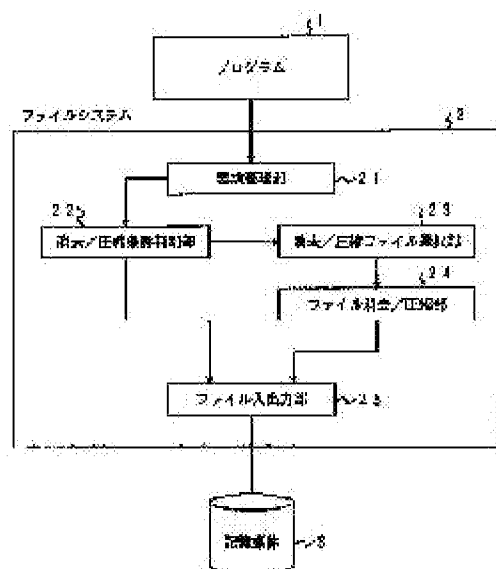
Priority number : **07219006** Priority date : **28.08.1995** Priority country : **JP**

(54) COMPUTER SYSTEM AND FILE MANAGING METHOD USED BY THE SAME

(57)Abstract:

PROBLEM TO BE SOLVED: To improve the utilization efficiency of a limited physical storage medium by automatically generating a free area on the storage medium for storing files.

SOLUTION: A file system 2 is judged to require automatic file erasure when the free area of the recording medium 3 decreases below a specific value or when the free area of the recording medium 3 is smaller than the file size of a file to be written. Once the necessity of the automatic file erasure is judged, an erased file selection part 23 selects a file to be erased according to the priority of the respective files stored on the recording medium 3. Then a file erasure part 24 follows a procedure for erasing the selected file. Therefore, a free area is automatically generated on the storage medium for storing the file even when a user does not generate an erasure request plainly, and the limited storage capacity of the physical storage medium 3 can efficiently be utilized.



* NOTICES *

JPO and INPIT are not responsible for any damages caused by the use of this translation.

- 1.This document has been translated by computer. So the translation may not reflect the original precisely.
- 2.**** shows the word which can not be translated.
- 3.In the drawings, any words are not translated.

DETAILED DESCRIPTION

[Detailed Description of the Invention]

[0001]

[Field of the Invention]This invention relates to the file management method used by the computer system and its system.

[0002]

[Description of the Prior Art]Generally, the data permanently owned with a computer system is stored in a nonvolatile storage medium like a disk in the logical unit of a file. The software which assigns a logical data unit called a file is called a file system to each memory part of a physical nonvolatile storage medium.

[0003]Since a file system utilizes a physical storage and stores data, there is a limit in a storage capacity. Although he wanted to newly create a file, when free space was lost, free space needed to be made from the former by specifying clearly the file which the user itself should eliminate and executing the program of file erasure. The program specified the file to the operating system and advanced the elimination demand, and when this was received, the operating system cancelled the applicable file memorized by the file system, and was making free space. That is, when the size of the file stored in a file system increases and free space lost or decreases. The file which a user should eliminate had to be chosen and free space had to be made by performing an elimination demand to an operating system clearly, and erasing a file by starting a program.

[0004]

[Problem(s) to be Solved by the Invention]As mentioned above, in the former, when there was not sufficient free space for a storage, the user itself generated the elimination demand clearly and had to eliminate the file.

[0005]For this reason, the directions of the program for performing operation of choosing the file which a user should eliminate, or eliminating had to be learned, and it had become a cause which worsens user-friendliness of a computer.

[0006]When data was transmitted from the exterior via a network etc., unless the user realized that free space is lost, data transmitting could not be stored in the file system, but it had become a cause by which reception of data went wrong as a result.

[0007]This invention is made in view of such a point, and is a thing.

The purpose is to provide the storage which stores a file with the computer system and file management method which can use efficiently for an automatic target the storage capacity of the physical storage

which enabled it to secure required free space and was restricted, even if it does not generate an elimination demand etc.

[0008]

[Means for Solving the Problem] This invention is characterized by that a computer system which manages file input and output over a recording medium which can save two or more files comprises: A means to distinguish whether it is based on free space of said recording medium, and there is any necessity of eliminating or compressing a file of said recording medium.

The file automatic deletion / compression means which performs procedure for choosing a file of elimination or a compression object automatically from files stored in said recording medium, and eliminating or compressing it when it is judged by this discriminating means that there is the necessity for file erasure or compression

[0009] In this computer system, it is judged whether automatic deletion of a file or compression is performed according to a size of free space of a recording medium. In this case, for example, when free space of a recording medium becomes below a predetermined value, or when there is less free space of a recording medium than a file size for writing, it is judged that free space secured processing by file automatic deletion or compression needs to be performed. If it is judged that free space secured processing by file automatic deletion or compression is required, a file of elimination or a compression object will be automatically chosen from files stored in a recording medium. And procedure for eliminating or compressing the selected file is performed. In procedure of this file erasure/compression, by publishing a file erasure demand or a file compression demand to the file erasure / compression by the file automatic deletion / the compression means itself, or an operating system, File erasure by the operating system is performed. Therefore, even if a user generates neither an elimination demand nor a compression demand clearly, free space can be automatically generated now to a storage which stores a file, and it becomes possible to use efficiently a storage capacity to which a physical storage was restricted. Therefore, it is not necessary to learn directions of a program for choosing a file which a user should eliminate, and a file which should be compressed, or eliminating or compressing, and a user-friendly computer system can be realized. When data is transmitted from a network or other computers, even if a user does not make free space, cannot lose data transmitting and it can be saved.

[0010] As for discrimination processing of whether to perform free space secured processing by file automatic deletion or compression, it is preferred to carry out periodically, when a file write request is published from a user program etc. It becomes possible to secure free space optimal thereby always, and a new file written in and specified by user program can prevent fault that a storage capacity is insufficient, and cannot save or download of a file from a network etc. goes wrong.

[0011] As for selection of elimination or a compression object file, it is desirable to choose an erasing object file and a compression object file according to a priority of a file in consideration of a size of influence which it has on a user by the file erasure or file compression. Here, a priority of a file is an index which shows a user what it has influence of by the file erasure or compression. Since it is decided by judgment of the user itself who is using the computer system which file to be important, as a priority, It is preferred to use input-and-output hysteresis information about priority information beforehand specified by user for every file, existence of file read-out of each file from a recording medium, the number of times of read-out, the last read-out time, etc.

[0012]The file name itself given to the file at the time of preservation of a new file may perform specification of a priority by a user, and it may be given to a file by making a parameter for exclusive use different from a file name into a priority. By using priority information such specified by a user, elimination file selection based on the user's itself importance judgment becomes possible. A file extension child can also be used as a priority and priority attachment according to a kind of files, such as a program file, a system file, a text file, a compressed file, and a graphics file, a relation with a program treating that file, etc. can be performed in this case.

[0013]When an erasing object file or a compression object file which a priority of all the files is higher than a reference value which chooses an elimination file and a compressed file, and corresponds does not exist, The reference value may be changed dynamically or a low file of a priority may be most chosen as an erasing object file or a compression object file in a file exceeding these reference values.

[0014]A file generally referred to by user recently like a file outputted and inputted frequently can be judged to be a file with importance high for the user. Although a file which a user has already referred to for data files downloaded from a network etc., such as news and a weather report, once on the other hand may erase, a use that data which has not been referred to yet must not be erased is also considered. Therefore, by choosing an elimination file based on an input-and-output history of a file, especially a history of file reference about read-out of a file, It can avoid, eliminating a file with great influence which it has on a user by elimination, i.e., a file it is expected to be to be read from now on, if possible.

[0015]A bigger file of size may eliminate preferentially, using a file size as a priority. An availability required only of file erasure of the thereby more small number is securable.

[0016]Files with a different form which the computer system of this invention can change mutually in a file stored in a storage. for example, a text file of an identical content from which only a file format differs. File content matching information which shows relation between a data file, its compression data file, etc. is held, with reference to the file content matching information, priority is given to a file which can be restored as an erasing object file, and it is chosen. Thereby, a file can be eliminated efficiently, without affecting a user at all. Especially when performing file erasure, it is suitable, but this composition can be used also about a case where file compression is performed.

[0017>About a file which has reference relation among other files like a hypertext. Since influence of [when the file is eliminated or compressed] becomes large so that there are many references from other files, It is preferred to detect the numbers of references, such as the number of other files which are referring to the file for each file stored in a recording medium, and to choose a file of an erasing object from files stored in said recording medium according to the detected number of references.

[0018]In addition to the above means, a computer system of this invention notifies a user of a file selected by an elimination file selection means, After obtaining elimination permission, by using combining suitably a means to eliminate the file etc., automatic file erasure with less influence on a user is realizable.

[0019]A computer system of this invention is provided with both sides of a file automatic deletion means and a file automatic compression means, and they are selectively used for it. Usually, although there is less influence which it has on a user than a case where a direction of file compression performs file deletion, file deletion can secure much free space from file compression easily. For this reason, more efficient file management becomes possible by using compression and elimination selectively by explicit specification by a user, etc., or, for example, choosing compression and elimination automatically based on free space size of a recording medium, etc.

[0020]This invention is provided with both sides of a file automatic deletion means and a file automatic compression means, Until a file which may be compressed is lost (for example, all the files are ending with compression, or.) Or when only a file like a file decided by a directory, file extension child, etc. fixed as a file which must not be compressed beforehand remains etc., When there is no file which may give priority to file automatic compression, and may perform and curtail it, file automatic compression is performed. By proper use of such compression and elimination, influence which it has on a user can be lessened more.

[0021]The optimal free space reservation is attained by changing which both sides of a compression object file and an erasing object file shall be chosen on each standard, and shall be performed between file erasure and compression according to a relation of a selected file.

[0022]

[Embodiment of the Invention]Hereafter, the embodiment of this invention is described with reference to drawings.

[0023]The composition of the file system used with the computer system concerning a 1st embodiment of this invention is shown in drawing 1. This file system 2 has a function for realizing either one of file automatic deletion and file automatic compression as a function for securing the availability of a recording medium automatically. This file system 2 is realized as a part of operating system. Although the computer system with which this file system 2 is applied comprises a usual computer with hardware resources, such as CPU, a memory, an auxiliary storage unit, and various I/O devices, The file automatic deletion / compression function of the file system 2, For example, since the capacity of auxiliary storage units for file memory, such as a portable terminal, is suitable for comparatively little electronic intelligence apparatus, below, the case where computer systems are portable electronic intelligence apparatus, such as PDA and a subnote PC, is illustrated and explained.

[0024]As shown in drawing 1 the electronic intelligence apparatus of this Embodiment 1, The user programs 1, such as an application program executed by CPU of the apparatus, It comprises the file system 2 which manages input and output of a file in response to the demand about a file operation from the program 1, and the nonvolatile storages 3, such as a hard disk, flash memory card, etc. which are used as an auxiliary storage unit for memorizing a file.

[0025]Although the operation about a file may be required by the command of OS level published according to the key input operations from [only from the user program 1] a user, when it sees from the file system 2 in both case, it will be required from other programs. Therefore, it will treat as a demand from the program 1 also about the demand from a user here.

[0026]The file system 2 is provided with the demand acceptance part 21, the elimination / compression condition discrimination section 22, the elimination / compressed file selecting part 23, the file erasure part 24, and the file input output section 25 like a graphic display for realization of the automatic free space secured function by file erasure or file compression.

[0027]The demand acceptance part 21 receives a file operation demand usual [, such as file read-out and writing,] from the user program 1. . [whether there is any necessity of elimination / compression condition discrimination section 22 being based on the free space of the recording medium 3, eliminating or compressing the file of the recording medium 3, and securing free space, and]That is, it is for distinguishing whether the conditions (elimination/compression condition) which perform free space secured processing by file erasure or compression are filling, Whenever the file operation demand of file writing etc. is published by the demand acceptance part 21, it is periodically started with a certain fixed

time interval, and discrimination processing of whether to fulfill elimination/compression condition is performed.

[0028]It is started when it is judged that elimination / compressed file selecting part 23 fulfills elimination/compression condition, The file which is the target of elimination or compression processing according to the priority of each of these files, etc. is chosen from the files (a data file and a program file are included) stored in the storage 3. Here, a priority is an index which shows a user what it has influence of when processing of elimination or compression is performed to the file, and it is a file with a high priority as a file with great influence which it has.

[0029]In file erasure / compression zone 24 processing either one of elimination or compression to the selected file and having an automatic deletion function, In performing processing which eliminates the selected erasing object file from the storage 3 and having an automatic compression function, it performs processing which compresses the selected compression object file. Here, with elimination of a file, "file deletion" may be used instead of a term "file erasure" synonymous with file deletion depending on an operating system.

[0030]When performing file erasure, may eliminate the substance of an erasing object file from the recording medium 3 (deletion), but. Only management data, such as a directory entry about the file, may be cancelled as well as the file erasure in the usual file system, and the technique of leaving the substance of a file as it is, or moving it to the storage area for exclusive use called what is called a "garbage can" may be used. Of course, when the external storage for backup can be used for a portable terminal, connecting, the erasing object file may be eliminated from the recording medium 3, and it may move to an external storage as a backup file.

[0031]The file structure managed with this file system 2 is shown in drawing 2.

[0032]At this embodiment, a file shall be managed by a tree structure. That is, it has two or more child nodes under a certain node, and the parent node seen from each node becomes settled uniquely. A node without a child node is a file and owns data. The node with a child node is called a directory. The directory like the top of a tree structure is called a route, and there is a parent node by itself.

[0033]In drawing 2, four files of file name "616.1" - "616.4" exist as a child node of the "news" directory, Signs that three files of file name "95.1" - "95.3" exist as a child node of the "schedule" directory are shown.

[0034]An example of the actual file management structure corresponding to drawing 2 is shown in drawing 3.

[0035]In drawing 3, the 1st field of each node expresses the kind of node. D is a directory and F is a file. The 2nd field is a name of a file or a directory. In the case of a directory, it is a pointer to a child node after the 3rd field. The slash in a figure shows an end and the value which is not usually used as a pointer is registered. For example, what is necessary is to use the address of a node, for an end to describe, to ** and just to use the minus 1 as a pointer.

[0036]In the case of a file, the value which shows the importance of the file is stored in the 3rd field. The value of this importance is used as a file priority mentioned above, and when choosing an erasing object file or a compression object file, it is referred to. In drawing 3, importance is expressed for the numerical values from 1 to 10, and it is so large that a numerical value is large. [of importance] The value of this importance is specified by the user at the time of creation/preservation of that file. The 4th field shows the size of actual data and the 5th field is a pointer to data.

[0037]The algorithm which shows the procedure of the file management processing of the file system 2 is shown in drawing 4.

[0038]First, the demand acceptance part 1 of the file system 2 receives the file operation demand from the user program 1 (Step S11). the file operation (refer to file read-out, file erasure, and a directory) specified by the file operation demand when the file operation demand was not a file write request (Step S12) -- usually -- a passage -- carrying out -- having (Step S13).

[0039]On the other hand, if the file operation demand from the user program 1 is a write request (Step S12), elimination / compression condition discrimination section 22 will judge whether predetermined elimination/compression condition are satisfied according to the availability of the storage 3 (Step S14). In this case, the file size by which the write request was carried out, and the present availability of the storage 3 are measured, and it will be distinguished, if elimination/compression condition is satisfied when the size of free space is smaller than the file size by which the write request was carried out. It may distinguish, if the present availability of the storage 3 is below default value and elimination/ compression condition will be satisfied regardless of the file size by which the write request was carried out.

[0040]The value of the present availability of the storage 3 is held within the file system 2. That is, whenever the file system 2 performs file writing, it subtracts the value of the full storage capacity of the storage 3 assigned to the file system 2 by the file size, it goes, and holds the result as the present availability of the storage 3.

[0041]If there is sufficient availability and the conditions of elimination or compression are not satisfied (Step S15), the writing of the demanded file will be performed immediately (Step S16). If there is not sufficient availability on the other hand and elimination/compression condition is satisfied (Step S15), file selection processing will be performed by elimination / compressed file selecting part 23 (Step S17).

[0042]Namely, as for elimination / compressed file selecting part 23, according to the priority (importance of drawing 3) given to these each file, the value of importance chooses the file below a reference value in the file stored in the storage 3. And the selected file is eliminated or compressed by file erasure / compression zone 24 (Step S18).

[0043]Then, repeat execution of the processing of Step S17 and S18 is carried out until discrimination processing of whether the conditions of elimination/compression are satisfied again is performed by elimination / compression condition discrimination section 22 (Step S14) and it stops materializing by it. And when the conditions of elimination/compression stopped satisfying, file writing processing of Step S16 is performed.

[0044]The example of the user program 1 in the case of performing a file write request in the demand acceptance part 21 is shown in drawing 5.

[0045]First, generation (file creation) of a new file is required by the program 1. In this case, as shown in (1) of drawing 5, a file name ("/schedule/95.6") is specified, and the system call for new file generating is sent to the file system 2. A success of file generating will return a file identification child (fd) to the program 1 from the file system 2. Next, the write request to a file is performed by the program 1. (2) is the demand writing out 256 bytes which begins from the position in the program specified by the variable buf to the file fd specified by the file identification child. That is, a write request writes in with the file for writing, specifies a number of bytes, and is performed. The demand acceptance part 21 receives this demand, and obtains the file and the write-in number of bytes for writing.

[0046]Next, it is distinguished whether elimination/compression condition as which elimination /

compression condition discrimination section 22 was determined beforehand are satisfied. In this case, as mentioned above, when the free space of the storage 3 is less than the file size by which the write request was carried out, or when free space is less than the size defined beforehand, it is judged that elimination/compression condition was satisfied.

[0047]The example of the distinction method of the latter by elimination / compression condition discrimination section 22 is shown in drawing 6.

[0048]In this discrimination processing, file erasure or compressive availability secured processing is repeated until it is less than another constant value SL, when the usage rate of the storage 3 by the file system 2 is over the constant value SU and a write request is received. The usage rate of the storage 3 will increase, if file writing is performed, and if elimination, partial deletion, or compression is performed, it will decrease. In drawing 6, since it is over SU when a write request is received in the time t1 and t2, signs that eliminated and even SL was reduced are shown.

[0049]The algorithm corresponding to the elimination discrimination processing of drawing 6 is shown in drawing 7.

[0050]Elimination / compression condition discrimination section 22 calculates the present usage rate SR of the storage 3 (Step S21), and investigates whether this is more than SU (Step S22). If it is $SR \geq SU$, elimination / compressed file selecting part 23 will be performed, and the file which should be eliminated or compressed will be chosen (Step S23). Subsequently, after the selected file is eliminated or compressed by file erasure / compression zone 24, usage rate SR is again calculated by elimination / compression condition discrimination section 22 (Step S24). And it is investigated by elimination / compression condition discrimination section 22 whether it is $SR \geq SL$, Step S23 and S24 wind until SR is less than SL, and it is *****. Generally it is $SL < SU$.

[0051]When elimination/compression condition was not satisfied and elimination/compression discrimination section 22 distinguishes, a write request is usually processed at a passage. That is, after checking write-in authority etc., if satisfactory, it will write in.

[0052]When elimination/compression condition was satisfied and elimination/compression discrimination section 22 distinguishes, elimination / compressed file selecting part 23 chooses the file which should be eliminated or compressed, but many algorithms can be considered for this selection.

[0053]For example, a file with the lowest importance to one of the files which exist in a file system is chosen. When there are two or more files of the minimum [importance], arbitrary one may also be chosen, the large file of a file size may also be most chosen in it, and the file of the file size minimum may also be chosen. Or the file of the minimum file size may also be chosen within limits which are not less than the file size by which the write request is carried out.

[0054]To use the object of elimination or compression about a specific file. Elimination / compressed file selecting part 23 is made for the file of the importance exceeding a certain value not to consider it as elimination or a compression object, and should just give the importance exceeding the set-up value to a file not to make into elimination or a compression object. For example, if the file exceeding the importance 9 shall not consider it as the object of elimination or compression, "95.1", "95.2", and "95.3" will not be made into the object of elimination or compression among the files shown in drawing 3.

Thereby, a specific file can be removed from elimination or a compression object. This value is not fixed and it may decide to be able to change.

[0055]Even when it is distinguished by elimination / compression condition discrimination section 22 that elimination or compression is required, elimination or a compression object file may be unable to be

chosen by elimination / compressed file selecting part 23. In this case, it is preferred to cope with it as follows, for example.

[0056](1) Elimination / compressed file selecting part 23 returns to the program 1 that there is no applicable file, or notifies a user of it, and does not carry out elimination or compression of a file.

[0057](2) If a file cannot be chosen when the method that elimination / compressed file selecting part 23 does not choose the file exceeding the value set up beforehand of importance is taken, The low file of importance is most chosen out of the file of the importance exceeding the value which changes the preset value, and is made low, or was set up beforehand. It is preferred to carry out, after notifying a user of that in these cases and obtaining consent.

[0058]The system configuration of the electronic intelligence apparatus concerning a 2nd embodiment of this invention is shown in drawing 8.

[0059]Here, elimination/compression informing part 26 is added to the file system 2, and it is constituted so that a user may be notified of a file with this selected as elimination or a compression object.

[0060]That is, the file selected by elimination / compressed file selecting part 23 is passed to elimination/compression notice program 4 by elimination/compression informing part 26, and a user is notified of it. Thereby, the file of a user can know [automatic deletion or being compressed, and] which file was eliminated or compressed. Before the file erasure by file erasure / compression zone 24 or compression or the back is also available for the timing of the notice by elimination/compression informing part 26.

[0061]The system configuration of the electronic intelligence apparatus concerning a 3rd embodiment of this invention is shown in drawing 9.

[0062]Here, in addition to the composition of Embodiment 2, the yes or no acceptance part 27 is formed in the file system 2.

[0063]That is, the file selected by elimination / compressed file selecting part 23 is passed to elimination/compression notice program 4 by elimination/compression informing part 26, and a user is notified of it. A user inputs whether elimination or compression is accepted to elimination/compression notice program 4, or it does not accept. This yes or no is received by the yes or no acceptance part 27, and is inputted into elimination / compressed file selecting part 24. This file will be eliminated or compressed if a user accepts execution of the free space secured processing by elimination or compression of that file. If it does not accept, elimination / compressed file selecting part 23 chooses another file.

Elimination and compression are performed, after neither elimination nor compression is made and obtaining a user's permission, while a user did not know, also although this called it the low file of importance. The flow chart corresponding to drawing 9 is shown in drawing 10.

[0064]First, the demand acceptance part 1 of the file system 2 receives the file operation demand from the user program 1 (Step S31). the file operation (refer to file read-out, file erasure, and a directory) specified by the file operation demand when the file operation demand was not a file write request (Step S32) -- usually -- a passage -- carrying out -- having (Step S33).

[0065]On the other hand, if the file operation demand from the user program 1 is a write request (Step S32), elimination / compression condition discrimination section 22 will judge whether predetermined elimination/compression condition are satisfied according to the availability of the storage 3 (Step S34). In this case, as mentioned above, the file size by which the write request was carried out, and the present availability of the storage 3 are measured, and it will be distinguished, if elimination/compression condition is satisfied when the size of free space is smaller than the file size by which the write request was carried out. It may distinguish, if the present availability of the storage 3 is below default value and

elimination/compression condition will be satisfied regardless of the file size by which the write request was carried out. Usage rate SR mentioned above can be used as free space.

[0066]If there is sufficient availability and elimination/compression condition is not satisfied (Step S35), the writing of the demanded file will be performed immediately (Step S36). If there is not sufficient availability on the other hand and elimination/compression condition is satisfied (Step S35), the selection process of elimination or the compression object file by elimination / compressed file selecting part 23 will be performed (Step S37). That is, elimination / compressed file selecting part 23 investigates a priority (importance of drawing 3) in order about each file stored in the storage 3, and the value of importance chooses the file below a reference value.

[0067]And the selected file is passed to elimination/compression notice program 4 by elimination/compression informing part 26, and a user is notified of it (Step S38, S39). A user inputs whether it accepts performing processing of elimination or compression to the file to elimination/compression notice program 4, or it does not accept. This yes or no is received by the yes or no acceptance part 27 (Step S40), and is inputted into elimination / compressed file selecting part 24. If a user accepts file erasure and file compression (Step S41), this file will be eliminated or compressed by file erasure / compression zone 42 (Step S42). If it does not accept, again, Steps S37-S41 are performed, and elimination / compressed file selecting part 23 chooses another file.

[0068]Other examples of priority management of each file are shown in drawing 11.

[0069]That is, although the field of importance was provided into each file and the value which shows the importance of a file in it was stored in drawing 3, the table which consists of a name of a file and a pair of importance may be created like drawing 11, and it may be stored as a separate file etc. This table is updated with generation of a file, deletion, and change of importance.

[0070]A priority may be managed by the file name itself which a user gives to the file at the time of creation/preservation of a new file. A file extension child can also be used as a priority and priority attachment according to the kind of files, such as a program file, a system file, a text file, a compressed file, and a graphics file, a relation with the program treating that file, etc. becomes possible in this case.

[0071]As mentioned above, although Embodiments 1-3 were described supposing the case where the file system 2 is built into the operating system as a file management section, The file automatic deletion / compression function of the file system 2 are also realizable as a program which moves by user levels outside an operating system. The system configuration example of the portable terminal in that case is shown in drawing 12.

[0072]Namely, in drawing 12, the file automatic deletion / compression function of the file system 2 explained by Embodiments 1-3 are mounted as the file management program 6 which operates out of an operating system, It operates separately from the general user program 1, and the demand from the direct general program 1 is not received. When it sees from the operating system 5, the file management program 6 is visible to the general program 1 and the same user program.

[0073]The functional constitution of the file management program 6 is shown in drawing 13.

[0074]Although elimination / compression condition discrimination section 31 in drawing 13, the elimination / compressed file selecting part 32, and the file erasure/compression zone 33 are equivalent to elimination / compression condition discrimination section 22 of drawing 1, the elimination / compressed file selecting part 23, and the file erasure/compression zone 24, respectively, The input output section 34 of drawing 13 functions as an interface with the operating system 5.

[0075]If elimination/compression condition is satisfied by elimination / compression condition discrimination section 31 and it will be distinguished, elimination / compressed file selecting part 32 will be performed, and the file which should be eliminated or compressed will be chosen. File erasure / compression zone 33 generates any of an elimination demand and a compression demand they are that the selected file should be eliminated or compressed. This is sent to the operating system 5 via the input output section 34 as a system call. The operating system 5 receives this, and eliminates or compresses a file. The concrete contents of processing performed by elimination / compression condition discrimination section 31, and the elimination / compressed file selecting part 32 are the same as the processing explained by drawing 6 and drawing 7.

[0076]As explained above, according to the file system 2 (the file management program 6 is included) used by the electronic intelligence apparatus of Embodiments 1-3. By providing either one of a file automatic deletion function or a file automatic compression function in it, Even if a user generates neither an elimination demand nor a compression demand clearly, free space can be automatically generated now to the storage 3 which stores a file, and it becomes possible to use efficiently the storage capacity to which the physical storage 3 was restricted. Therefore, even when there is little free space, it is not necessary to learn the directions of the program for choosing the file which a user should eliminate or eliminating, and a user-friendly computer system can be realized. When data is transmitted from a network or other computers, even if a user does not make free space, cannot lose data transmitting and it can be saved.

[0077]In the above explanation, when a file write request was published from the user program 1, the existence of formation of elimination/compression condition was judged, but this decision processing may be periodically performed with a certain time interval. This is suitable for the composition of the user program 1 and drawing 13 which is independently.

[0078]Next, a 4th embodiment of this invention is described with reference to drawing 14 thru/or drawing 21.

[0079]Although the value of the importance decided to have mentioned above fixed for every file by Embodiments 1-3 was used as a priority for elimination / compressed file selection, In Embodiment 4, it is the composition which uses the operating condition of the file by the history about input and output of each file, i.e., a user, as a priority for elimination / compressed file selection. Using such a priority is based on the following reasons.

[0080]That is, with a computer system, the file referred to frequently and the file which is not referred to not much frequently usually exist. When the file referred to frequently is erased, a possibility that the problem that a file is not found even if a user is going to refer to the file again will arise is high. If similarly the file referred to frequently is compressed, an expansion process will be needed for the degree of read-out of the file, and the problem that the time which file read-out takes becomes long will occur.

[0081]Since it can assume that it is a file with importance low for a user which may once be read contrary to this about document files, such as news downloaded from the network etc., Although the file which the user already referred to once may erase, there is also a use that the file which has not been referred to yet must not erase.

[0082]Therefore, in Embodiment 4, in order to correspond to such a situation, the hysteresis information about read-out of a file is used as a priority. Hereafter, the functional constitution of Embodiment 4 is explained.

[0083]The electronic intelligence apparatus of this Embodiment 4 like the system configuration of Embodiments 1-3 mentioned above The user programs 10, such as an application program, It comprises the file system 20 which manages input and output of a file in response to the demand about a file operation from the program 10, and the nonvolatile storages 30, such as a hard disk, flash memory card, etc. which are used as an auxiliary storage unit for memorizing a file.

[0084]The file system 20 comprises the file input output section 201, the file read section 202, the file read-out history preparing part 203, the elimination / compressed file selecting part 204, and the file erasure/compression zone 205.

[0085]It is reported that the file read section 202 read the file from the file storing part 31 of the storage 30 via the file input output section 201 at the time of read-out of a file, and also the file name of an applicable file and read-out of the file were in the file read-out history preparing part 203. The file read-out history preparing part 203 creates the file read-out hysteresis information which shows the existence of file read-out, the number of times of read-out, the last read-out time, etc. for every file according to the notice from the file read section 202. This file read-out hysteresis information is stored in the file read-out history storage part 32 of the storage 30 via the file input output section 201.

[0086]Elimination / compressed file selecting part 204 refers to the file read-out hysteresis information of the file read-out history storage part 32 via the file input output section 201, The file assumed not to be important for a user is chosen as elimination or a compression object file from the files stored in the file storing part 31, and the file name is notified to file erasure / compression zone 205. Like Embodiments 1-3, elimination / compressed file selection process of this elimination / compressed file selecting part 204 are periodically performed, when the file operation demand of file writing etc. is published from the program 10.

[0087]File erasure / compression zone 205 eliminates or compresses the selected file. In this case, elimination and compression of a file are performed by the same method as Embodiments 1-3.

[0088]Next, with reference to drawing 15 - drawing 17, file read-out hysteresis information is explained concretely.

[0089]In the example of drawing 15, file read-out hysteresis information is held as one of the file management information in the directory. File read-out hysteresis information is expressed by the flag of "0" or "1", it is shown that a flag "0" is a file which is not read, and it is shown that a flag "1" is a file which may be read.

[0090]In the example of drawing 16, not the flag that shows the existence of file read-out as file read-out hysteresis information but the value which shows the number of times of file read-out is used, and it is held at the directory entry of the file which each corresponds like drawing 15.

[0091]These drawing 15 or the file read-out hysteresis information of drawing 16 can also be saved as a read-out history file separately from a directory as shown in drawing 17. About the last read-out time of a file as well as the existence of file read-out, or the number of times of read-out, it is manageable. Since there are some which have the function to manage the last read-out time for every file depending on an operating system in itself, it may be used.

[0092]Next, with reference to drawing 18, the file automatic deletion / compression operations using a file read-out history are explained. Here, the case where give priority to the low file of referring frequency, and it deletes or compresses is assumed.

[0093]First, it is investigated whether whether there being any free space sufficient by the same

technique as Embodiments 1-3 for the file storing part 31 of the storage 30, and elimination/compression condition are satisfied (Step S51). If elimination/compression condition is satisfied, processing of elimination / compressed file selecting part 204 will be started. Namely, first elimination / compressed file selecting part 204, The read-out hysteresis information about one file is read from the file read-out history storage part 32 (Step S53), and the file is not read, or it is investigated whether the number of times of read-out is a file below a predetermined value (Step S54). It is not read, or if the number of times of read-out is a file below a predetermined value, elimination / compressed file selecting part 204 will tell the file name to file erasure / compression zone 205, and will make it eliminate or compress (Step S55).

[0094]Then, it is investigated again whether elimination/compression condition is satisfied (Step S51), and Steps S53-S55 are repeated until elimination/compression condition stops satisfying, it reads about all the files and read-out of hysteresis information is completed.

[0095]Thus, it can prevent it being read frequently and read also from this from if possible eliminating or compressing the file expected by determining the file which performs elimination and compression using a read-out history. Therefore, the influence which it has on a user by file erasure/compression can be reduced substantially.

[0096]Other examples of file read-out hysteresis information are shown in drawing 19.

[0097]Here, the last read-out date information of each file is used as file read-out hysteresis information. In this case, at the time of read-out of a file, the file read section 202 reads a file from the file storing part 31, and also the file name of an applicable file is notified to the file read-out history preparing part 203. The file read-out history preparing part 203 makes the notified file name and the present time correspond, and is memorized to the file read-out history storage part 32.

[0098]Next, with reference to drawing 20, the file automatic deletion / compression operations using a file read-out history are explained. Here, fundamentally, the last read-out time gives priority to a older file, and is deleted or compressed.

[0099]First, it is investigated whether whether there being any free space sufficient by the same technique as Embodiments 1-3 for the file storing part 31 of the storage 30, and elimination/compression condition are satisfied (Step S61). If elimination/compression condition is satisfied, processing of elimination / compressed file selecting part 204 will be started. Namely, first elimination / compressed file selecting part 204, The read-out hysteresis information about one file is read from the file read-out history storage part 32 (Step S63), and it is investigated whether it is a file in which the last read-out time is older than the beforehand fixed predetermined time (Step S64). If the last read-out time is a file older than predetermined time, elimination / compressed file selecting part 204 will tell the file name to file erasure / compression zone 205, and will make it eliminate or compress (Step S65).

[0100]Then, it is investigated again whether elimination/compression condition is satisfied (Step S61), and Steps S63-S65 are repeated until elimination/compression condition stops satisfying, it reads about all the files and read-out of hysteresis information is completed.

[0101]Thus, since the file referred to more by recording the last read-out time these days can be made not to eliminate or compress if possible, efficient free space reservation can be performed. If it is used combining the number of times of read-out which mentioned the last read-out time above, it will come excepting from elimination/compression object, or to be able to do about the number of times of read-out, or what is [the last read-out time] new at least, and still more efficient free space reservation will be attained.

[0102]Next, with reference to drawing 21, another example of the file automatic deletion / compression operations using the read-out hysteresis information (existence of read-out) of drawing 15 is explained. Here, the file which may be read is eliminated or compressed preferentially.

[0103]First, it is investigated whether there being any free space sufficient by the same technique as Embodiments 1-3 for the file storing part 31 of the storage 30, and elimination/compression condition are satisfied (Step S71). If elimination/compression condition is satisfied, processing of elimination / compressed file selecting part 204 will be started. That is, it is investigated first whether elimination / compressed file selecting part 204 is the files which read the read-out hysteresis information about one file from the file read-out history storage part 32 (Step S73) and from which the file may be read (Step S74). If it is a file which may be read, elimination / compressed file selecting part 204 will tell the file name to file erasure / compression zone 205, and will make it eliminate or compress (Step S75).

[0104]Then, it is investigated again whether elimination/compression condition is satisfied (Step S71), and Steps S73-S75 are repeated until elimination/compression condition stops satisfying, it reads about all the files and read-out of hysteresis information is completed.

[0105]Thus, the algorithm of eliminating or compressing preferentially the file which the user has already referred to once, Document files, such as news and a weather report, are downloaded to electronic intelligence apparatus, such as a portable terminal, through media, such as a telephone line, a network or an electric wave, It becomes possible to use efficiently the storage of the electronic intelligence apparatus which is effective especially in the use that a user refers to the they-downloaded file if needed, and has few storage capacities.

[0106]The system configuration of the electronic intelligence apparatus concerning a 5th embodiment of this invention is shown in drawing 22.

[0107]This system adds roughly the function which eliminates the information corresponding to the time of file writing in the file read-out history storage part 32 to the file system 20 of Embodiment 4. Namely, although the point of choosing the file which is the target of elimination or compression with reference to file read-out hysteresis information is the same as Embodiment 4, the file system 20 used by this Embodiment 5, In order to improve the reliability of the file read-out hysteresis information, the file writing part 206 is added. The erasing operation of the file read-out hysteresis information using this file writing part 206 is shown in drawing 23.

[0108]In [as shown in the flow chart of drawing 23] the time of file writing, The file writing part 206 notifies the file name of an applicable file for a file to the file storing part 31 with writing first at the file read-out history preparing part 203 (Step S81, S82). The file read-out history preparing part 203 eliminates or resets the file read-out hysteresis information of the file specified by the file name to an initial value (Step S83).

[0109]For example, in the case where the file read-out hysteresis information which shows the existence of read-out is being used as shown in drawing 24, When the writing about the file of the file name f3 is performed, the read-out hysteresis information corresponding to the file of the file name f3 is reset by "0", and the file is treated by this as a file which is not read.

[0110]Thus, by initializing file read-out hysteresis information applicable according to file writing, Although the data which the user already referred to once may be erased, in the use that the data which has not been referred to yet must not be erased, it takes into consideration whether the user read the data

after updating, and the file of an erasing object can be chosen.

[0111]The system configuration of the electronic intelligence apparatus concerning a 6th embodiment of this invention is shown in drawing 25.

[0112]The file system 20 used by this system, For realization of the free space secured function by file automatic deletion or file automatic compression, Like a graphic display, it has the file input output section 301, the file writing part 302, the file size information preparing part 303, the elimination / compressed file selecting part 304, and the file erasure/compression zone 305.

[0113]At the time of the writing of a file, the file writing part 302 writes a file in the file storing part 31 through the file input output section 301, and also the file name and file size of an applicable file are notified to the file size information preparing part 303. The file size information preparing part 303 generates the table showing correspondence with a file name and a file size, and memorizes it to the file size information storage part 33 through the file input output section 301. This file size information is also storable in a part of directory entry of each file. In the case of file erasure or compression, elimination / compressed file selecting part 304 obtains the size of each file with reference to the file size information storage part 33. And the file exceeding the decided value with size is chosen, and the file name of those files is told to file erasure / compression zone 305. File erasure / compression zone 305 eliminates the file of the file name obtained from elimination / compressed file selecting part 304 from the file storing part 31.

[0114]Next, with reference to drawing 26, the file automatic deletion / compression operations using file size information are explained. Here, the big file of size is eliminated or compressed preferentially.

[0115]First, it is investigated whether whether there being any free space sufficient by the same technique as Embodiments 1-3 for the file storing part 31 of the storage 30, and elimination/compression condition are satisfied (Step S91). If elimination/compression condition is satisfied, processing of elimination / compressed file selecting part 304 will be started. That is, first, elimination / compressed file selecting part 304 reads the size information about one file from the file size information storage part 33 (Step S93), and investigates whether the file size is over the reference size decided beforehand (Step S94). If it is a file with a larger file size than reference size, elimination / compressed file selecting part 304 will tell the file name to file erasure / compression zone 305, and will make it eliminate or compress (Step S95).

[0116]Then, it is investigated again whether elimination/compression condition is satisfied (Step S91), and Steps S93-S95 are repeated until elimination/compression condition stops satisfying and read-out of the size information about all the files is completed.

[0117]Thus, by eliminating or compressing a file with big size preferentially, The probability of occurrence of the problem that it is compressed and the read-out takes time or it can stop the number of the file eliminated or compressed and the file which a user tried to read is already eliminated, in order to vacate the capacity of the file storing part 31 can be reduced.

[0118]Processing in which a file in this way with big size is eliminated or compressed preferentially can also be used combining the file read-out hysteresis information explained by Embodiments 4 and 5. In this case, it is preferred to first adopt how two or more candidate files are selected, and size eliminates or compresses a big file preferentially in it by referring to file read-out hysteresis information.

[0119]The system configuration of the electronic intelligence apparatus concerning a 7th embodiment of this invention is shown in drawing 27.

[0120]The file system 20 used by this system, For realization of the free space secured function by file

automatic deletion or compression, it has the file input output section 401, the file conversion part 402, the relation part 403 corresponding to a file content, the elimination / compressed file selecting part 404, and the file erasure/compression zone 405 like a graphic display.

[0121]The file conversion part 402 reads the contents of a certain file fa from the file storing part 31, and after it adds conversion (compression/extension of file data, conversion of document file form, etc.) to the contents, it stores the file after conversion as another file fb in the file storing part 31. In this case, the file conversion part 402 notifies the relation preparing part 403 corresponding to a file content that it is a file from which the files fa and fb have convertible contents mutually, and only a data format differs.

[0122]Based on the notice from the file conversion part 402, the relation preparing part 403 corresponding to a file content, File content pair relation ***** which shows the relation with contents mutually convertible in the file stored in the file storing part 31 of files is generated, and it is stored in the relationship storing part 34 corresponding to a file content.

[0123]As file content pair relation *****, the file extension child etc. who are registered into the directory entry of each file can also be used. Namely, if the same file name as the file fa is given to the compressed file fb obtained by compressing a certain file fa, for example and the information which shows data compression format as an extension which shows the file type is given, It can recognize that they are the file fa and a file with the contents which fb can change mutually from the file fa, and the file name of each fb and the relation of an extension. Therefore, if this method is adopted, the relationship storing part 34 corresponding to a file content is realizable as some directories.

[0124]First, with reference to the relationship storing part 34 corresponding to a file content, in the case of file erasure or compression, elimination / compressed file selecting part 404 chooses the file which can be restored from the contents of other files, and tells the file name of the file to file erasure / compression zone 405 at it. File erasure / compression zone 405 eliminates or compresses the file of the file name obtained from elimination / compressed file selecting part 404 from the file storing part 31.

[0125]Next, with reference to drawing 28, the file automatic deletion / compression operations using the relation memory information corresponding to a file content are explained. Here, the file which the contents can restore by conversion from other files is eliminated or compressed preferentially.

[0126]First, it is investigated whether whether there being any free space sufficient by the same technique as Embodiments 1-3 for the file storing part 31 of the storage 30, and elimination/compression condition are satisfied (Step S101). If elimination/compression condition is satisfied, processing of elimination / compressed file selecting part 404 will be started. That is, first, elimination / compressed file selecting part 404 reads the information about one file from the relationship storing part 34 corresponding to a file content (Step S103), and investigates whether the file is a file which can be restored from other file contents (Step S104). If it is a file which can be restored, elimination / compressed file selecting part 404 will tell the file name to file erasure / compression zone 405, and will make it eliminate or compress (Step S105).

[0127]Then, it is investigated again whether elimination/compression condition is satisfied (Step S101), and Steps S103-S105 are repeated until elimination/compression condition stops satisfying and read-out of the correspondence relevant information about all the files is completed.

[0128]Thus, the file which can restore the contents by conversion from other files is eliminated or compressed preferentially, Generating of the problem that the control kept from performing processing of elimination or compression to the file whose restoration is impossible is effective especially when the automatic file erasure for free space reservation is adopted, and the file which a user tried to read is

already eliminated and cannot read it can be reduced. Processing in which the file which can be restored in this way is eliminated preferentially can also be used combining the file read-out hysteresis information explained by Embodiments 4 and 5. In this case, it is preferred to first adopt the method of selecting two or more erasing object candidates, and eliminating preferentially the file which can be restored in it by referring to file read-out hysteresis information.

[0129]The system configuration of the electronic intelligence apparatus concerning an 8th embodiment of this invention is shown in drawing 29.

[0130]The file system 20 used by this system, For realization of the suitable free space automatic secured function for the file which has reference relation among other files like a hypertext, Like a graphic display, it has the file input output section 501, the reference relation read section 502, the elimination / compressed file selecting part 503, and the file erasure/compression zone 504.

[0131]In the case of file erasure or compression, first, the reference relation read section 502 starts operation with the directions from elimination / compressed file selecting part 503, and the number of references from other files is detected about each file of the file storing part 31. The number of references analyzes the contents of each of other file, and is detected by counting the number of the pointers to an applicable file. Elimination / compressed file selecting part 503 chooses little file from fixed numbers with the number of references from other files, and tells the file name of the file to file erasure / compression zone 504. File erasure / compression zone 504 eliminates the file of the file name obtained from elimination / compressed file selecting part 503 from the file storing part 31.

[0132]Next, with reference to drawing 30, the file automatic deletion / compression operations using the number of file references are explained. Here, little file of the number of reference in the file from other files is eliminated or compressed preferentially.

[0133]First, it is investigated whether whether there being any free space sufficient by the same technique as Embodiments 1-3 for the file storing part 31 of the storage 30, and elimination/compression condition are satisfied (Step S111). If elimination/compression condition is satisfied, processing of elimination / compressed file selecting part 503 will be started. That is, elimination / compressed file selecting part 503 reads one file from the file storing part 31 (Step S113), and makes the reference relation read section 502 count the number of references from other files about the file first. The reference relation read section 502 analyzes the contents of each of other file, by counting the number of the pointers to an applicable file, detects the number of references and passes the result to elimination / compressed file selecting part 503. It is investigated whether elimination / compressed file selecting part 503 has few references corresponding to the file than the reference value defined beforehand (Step S114). If the number of references from other files is a file less than a reference value, elimination / compressed file selecting part 503 will tell the file name to file erasure / compression zone 504, and will make it eliminate or compress (Step S115).

[0134]Then, Steps S113-S115 are repeated until it is investigated again whether elimination/compression condition is satisfied (Step S111), elimination/compression condition stops satisfying and the number of references from other files about all the files is counted.

[0135]Thus, by the reference from other files eliminating or compressing little file preferentially, and saving a file with much reference in a form as it is, When file erasure is especially adopted for a free space secured function, generating of the problem that the file which a user tried to read via the reference from other files is already eliminated can be reduced. Processing in which little file of the number of references is eliminated preferentially in this way can also be used combining the file read-

out hysteresis information explained by Embodiments 4 and 5. In this case, it is preferred to first adopt the method of selecting two or more erasing object candidates, and eliminating preferentially little file of the number of references from other files in it by referring to file read-out hysteresis information.

[0136]The file automatic deletion / compression function of the file system 20 of Embodiments 4-8 are also realizable like the case of the file system 2 of Embodiments 1-3 as a program which moves by user levels outside an operating system. In this case, only drawing 14, drawing 22, drawing 25, drawing 27, and the file input output section of each drawing 29 will be mounted in an operating system as a file management section, and other portions will be mounted as a file management program which operates out of an operating system.

[0137]As explained above, according to the electronic intelligence apparatus of Embodiments 4-8, by using the number of the correspondence relations and reference of the reference history, the size, and the contents of a file, etc. as a priority for elimination or compressed file selection, Little file automatic deletion of the influence which it has on a user rather than the case where only a static priority is used can be performed.

[0138]Also in the electronic intelligence apparatus of Embodiments 4-8, as Embodiments 2 and 3 explained, After notifying the user of the file with selected elimination / compressed file selecting part and obtaining permission of elimination/compression, it is preferred to use the file, combining suitably the function to eliminate or compress.

[0139]Next, a 9th embodiment of this invention is described.

[0140]The electronic intelligence apparatus of this Embodiment 9 like the system configuration of Embodiments 1-3 mentioned above The user programs 100, such as an application program, It comprises the file system 200 which manages input and output of a file in response to the demand about a file operation from the program 100, and the nonvolatile storages 300, such as a hard disk, flash memory card, etc. which are used as an auxiliary storage unit for memorizing a file.

[0141]The file system 200 is composition which has the both sides of a file automatic deletion function and a file automatic compression function, and uses selectively these file automatic deletion function and a file automatic compression function as a function for securing the availability of the recording medium 300 automatically.

[0142]The demand acceptance part 601 in which the file system 200 receives the demand from the program 100, The elimination file selection part 602 which chooses the file which should be eliminated, and the file erasure part 603 which eliminates a file, It consists of the compressed file selecting part 604 which chooses the file which should be compressed, the file compression part 605 which compresses a file, the file input output section 606 which outputs and inputs the file memorized by the storage 300, and the control section 607 which manages these.

[0143]Drawing 32 shows the file structure managed with the file system 200.

[0144]The 1st field of each node expresses the kind of node. D is a directory and F is a file. The 2nd field is a name of a file or a directory. In the case of a directory, it is a pointer to a child node after the 3rd field. A slanting line shows an end and should just use the value which is not used as a pointer. For example, what is necessary is to store the address of a node, for an end to describe, to ** and just to use the minus 1 as a pointer. In the case of a file, the time of a last update date is shown in the 3rd field, and the time of the last reference day is shown in the 4th field. It is shown that / is not referred to. The 5th field is a pointer to data.

[0145]Drawing 33 is a flow chart which shows the algorithm of a filing system. In this embodiment, the

time of the last reference day of a file is used as an attribute for choosing compression and an elimination file.

[0146]First, the demand acceptance part 601 of the file system 200 receives the file operation demand from the user program 100 (Step S201). the file operation (refer to file read-out, file erasure, and a directory) specified by the file operation demand when the file operation demand was not a file write request (Step S202) -- usually -- a passage -- carrying out -- having (Step S203).

[0147]On the other hand, if the file operation demand from the user program 100 is a write request (Step S202), the control section 607 will judge whether predetermined elimination/compression condition are satisfied according to the availability of the storage 300 (Step S204). In this case, the file size by which the write request was carried out, and the present availability of the storage 300 are measured, and it will be distinguished, if elimination/compression condition is satisfied when the size of free space is smaller than the file size by which the write request was carried out. It may distinguish, if the present availability of the storage 300 is below default value and elimination/compression condition will be satisfied regardless of the file size by which the write request was carried out.

[0148]If there is sufficient availability and the conditions of elimination or compression are not satisfied (Step S205), the writing of the demanded file will be performed immediately (Step S206). If there is not sufficient availability on the other hand and elimination/compression condition is satisfied (Step S205), it will be judged whether the compressible file judged that it may compress into the storage 300 by the control section 607 exists (Step S207). This judgment can be made when all the files, for example, investigate whether it is finishing [compression]. In this case, if the incompressible file remains, and it is judged that a compressible file exists and it does not remain, it is judged that a compressible file does not exist. The file which must be beforehand compressed by neither directory nor a file extension child, and the file which may be compressed are decided fixed by explicit specification by a user, etc., and it can also be judged whether a compressible file exists based on it.

[0149]If the compressible file judged that it may compress into the storage 300 remains, the compressed file selecting part 604, The oldest thing at the time of the last reference day (interpreted as the final reference time of the file which is not referred to being the oldest) is chosen as a compression object file in the file which may be compressed, and this is compressed by the file compression part 605 (Step S208, S209). Then, as long as the file which may be compressed remains until discrimination processing of whether the conditions of elimination/compression are satisfied again is performed (Step S204, S205) and it stops materializing, repeat execution of Step S208 and S209 is carried out.

[0150]On the other hand, when there is no file which may be compressed, the processing performed for free space reservation is shortly changed from file compression to file erasure.

[0151]In this case, the elimination file selection part 602 chooses the oldest thing at the time of the last reference day as an erasing object file in all the files, and this is eliminated by the file erasure part 603 (Step S210, S211).

[0152]Then, repeat execution of Step S210 and S211 is carried out until discrimination processing of whether the conditions of elimination/compression are satisfied again is performed (Step S204, S205) and it stops materializing.

[0153]Thus, since it is controlled by this embodiment so that file automatic compression processing gives priority to and performs file automatic erasing processing nearby, the influence which it has on a user by execution of the automatic secured function of free space can be suppressed less.

[0154]What is necessary is just to interpret it as the final reference time of the file which is not referred

to being the oldest. If this is repeated, the free space more than equivalent can be made to the file to write in.

[0155]When choosing the object file of compression and elimination, both used same attribute called final reference time here, but it is not necessarily this limitation. An attribute which is different so that it may say that compression chooses the oldest thing of final reference time, and elimination chooses what has a large file size may be used.

[0156]Drawing 34 is a flow chart which shows another algorithm of a filing system. In the example shown in drawing 33, after compressing all the incompressible files, for example, in order to eliminate, there is a possibility that the high file of a possibility of being used succeeding may also be compressed. A best policy does not compress the file used immediately consuming CPU time and electric power for compression of a file, and by returning to the state where there is no free space, immediately. It is better for the file with a high possibility of being used immediately to eliminate the low file of a possibility of being used rather, without compressing, even if it is in an incompressible state, in order to improve this. Since the judgment of whether to be used generally is difficult, time locality is assumed and it judges with a possibility that the thing which has the new time of the last reference day will be used being high in many cases. This is used also with this algorithm.

[0157]First, the demand acceptance part 601 of the file system 200 receives the file operation demand from the user program 100 (Step S301). the file operation (refer to file read-out, file erasure, and a directory) specified by the file operation demand when the file operation demand was not a file write request (Step S302) -- usually -- a passage -- carrying out -- having (Step S303).

[0158]On the other hand, if the file operation demand from the user program 100 is a write request (Step S302), the control section 607 will judge whether predetermined elimination/compression condition are satisfied according to the availability of the storage 300 (Step S304). In this case, the file size by which the write request was carried out, and the present availability of the storage 300 are measured, and it will be distinguished, if elimination/compression condition is satisfied when the size of free space is smaller than the file size by which the write request was carried out. It may distinguish, if the present availability of the storage 300 is below default value and elimination/compression condition will be satisfied regardless of the file size by which the write request was carried out.

[0159]If there is sufficient availability and the conditions of elimination or compression are not satisfied (Step S305), the writing of the demanded file will be performed immediately (Step S306). If there is not sufficient availability on the other hand and elimination/compression condition is satisfied (Step S305), the oldest thing at the time of the last reference day that is a file which may be compressed will be chosen, for example out of an incompressible file, it will be referred to as F_c , and the time of the last reference day of the F_c will be set to $t(F_c)$ (Step S307). Next, the old thing at the time of the last reference day is most chosen out of all the administration object files, it is referred to as F_d , and the time of the last reference day of the F_d is set to $t(F_d)$ (Step S307).

[0160]Next. [whether when setting the present time to t , the constant p which the ratio of $t(F_d)-t_l$ to $t(F_c)-t_l$ defined beforehand is exceeded, and] That is, elimination of F_d will be performed, if it is a file with it when [at which it is investigated whether $t(F_d)-t_l / t(F_c)-t_l > p$ is materialized (Step S309)] materialized that is, (Step S311). [remarkable F_d and] [older than the time of the last reference day of compression object file F_c] On the other hand, when the conditions of Step S309 are not satisfied, compression of compression object file F_c is performed (Step S310).

[0161]When using the last change time as an attribute of compression or elimination, the last change

time of the file made by compression shall just inherit the last change time of an incompressible state. When using the time of the last reference day as an attribute of compression or elimination, access to the attribute of the file which is made into the object of compression or elimination and which was performed for accumulating shall not regard reference, and it shall not be changed at the time of the last reference day.

[0162]Especially in above Embodiments 1-9, although the object of file management was not described, the object of management may be limited. . For example, only the file system of a certain drive becomes an administration object of the file concerned. . Only the specific file system in a certain drive becomes an administration object of the file concerned. Only the file from which only the bottom of a certain directory becomes an administration object of the file concerned and which a certain application uses may restrict the file which can serve as compression or a candidate for deletion using a controlling method, such as becoming an administration object of the file concerned. As mentioned above, the file management does not necessarily need to be a part of OS. It may realize as another module. Or when only the file which a certain application uses becomes an administration object of the file concerned, the application may perform the file management concerned.

[0163]Since the file management method of Embodiments 1-9 is realized as a program which can be executed by computer, By storing and distributing the program to recording media, such as a floppy disk and CD-ROM, it becomes possible only by installing the program in a computer from these media to manage the free space of secondary storage, such as a disk, simply.

[0164]

[Effect of the Invention]Since free space secured processing by file erasure or compression is automatically performed according to the free space of the storage which a file system manages according to this invention as explained above, It is not necessary to learn the directions of the program for choosing the file which a user should eliminate or eliminating, and a user-friendly computer system can be provided. When data is transmitted from the exterior, even if a user does not make free space, data transmitting is not lost but it can store in a file system. By eliminating using the reference history of a file, the correspondence relation of the contents, and the number of reference, a possibility that a user will refer to it can eliminate a low file now, and can improve the convenience of automatic deletion.

[Translation done.]

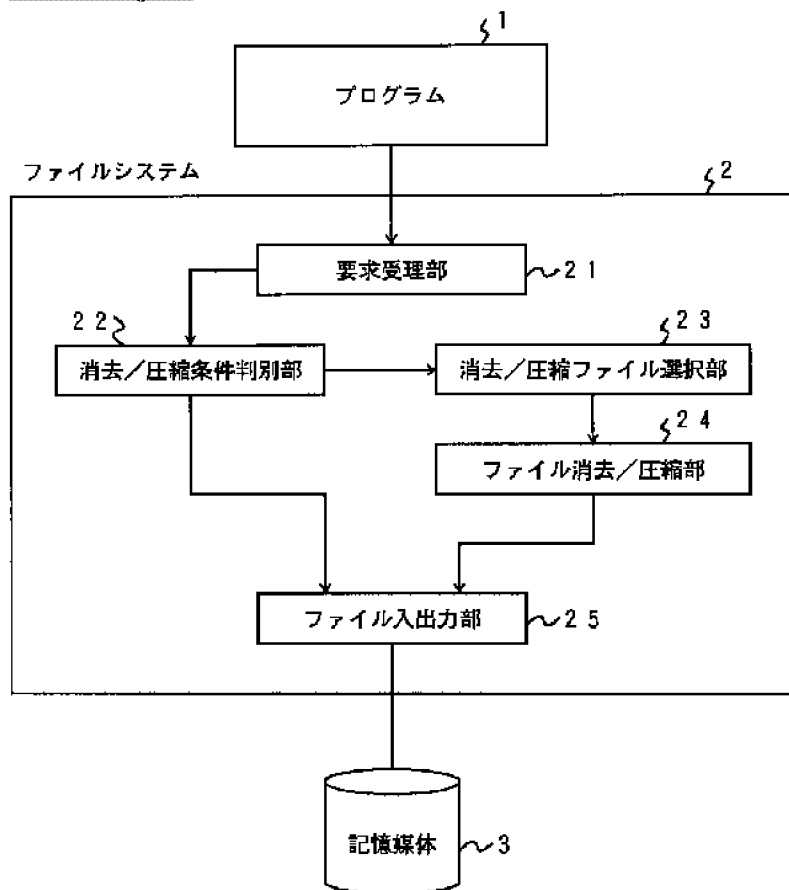
* NOTICES *

JPO and INPIT are not responsible for any damages caused by the use of this translation.

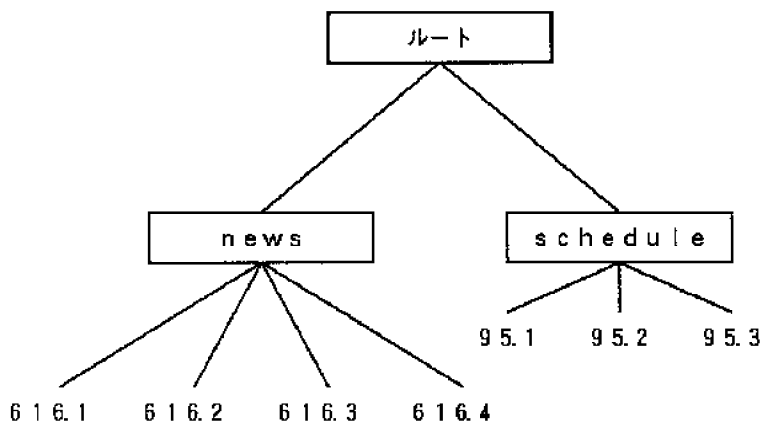
- 1.This document has been translated by computer. So the translation may not reflect the original precisely.
- 2.*** shows the word which can not be translated.
- 3.In the drawings, any words are not translated.

DRAWINGS

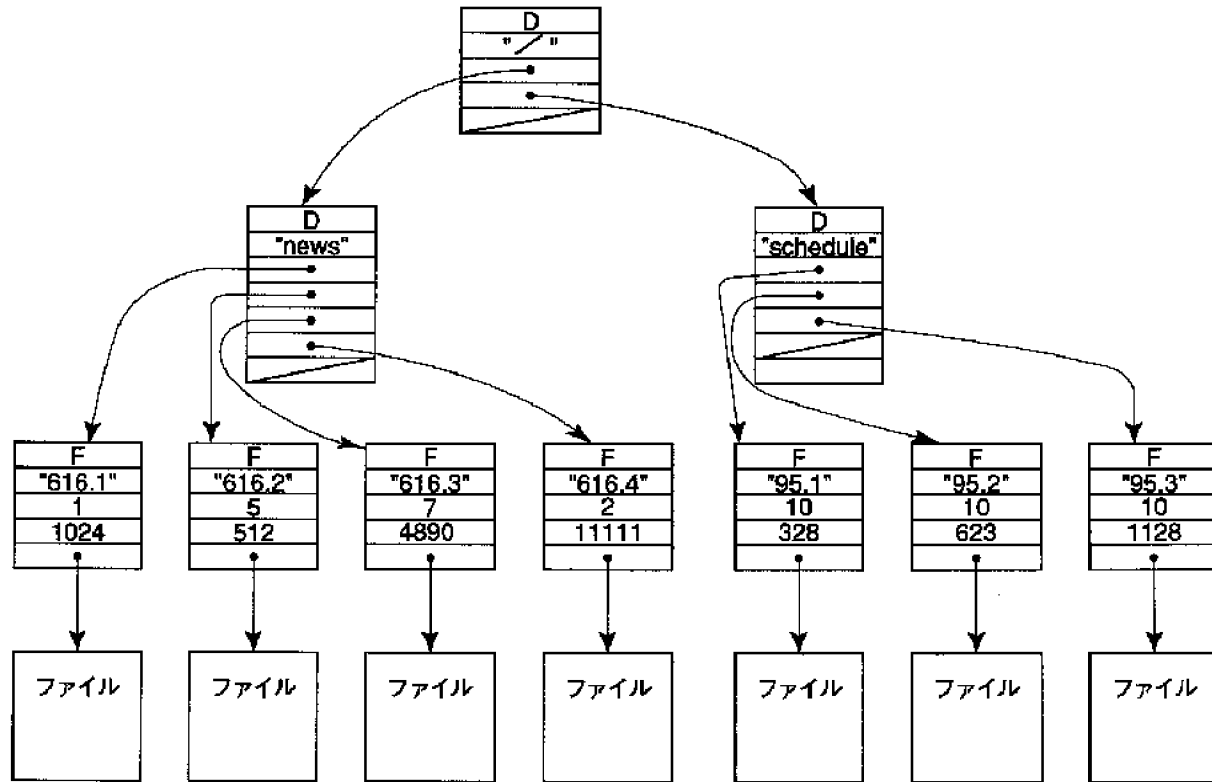
[Drawing 1]



[Drawing 2]



[Drawing 3]



[Drawing 5]

```

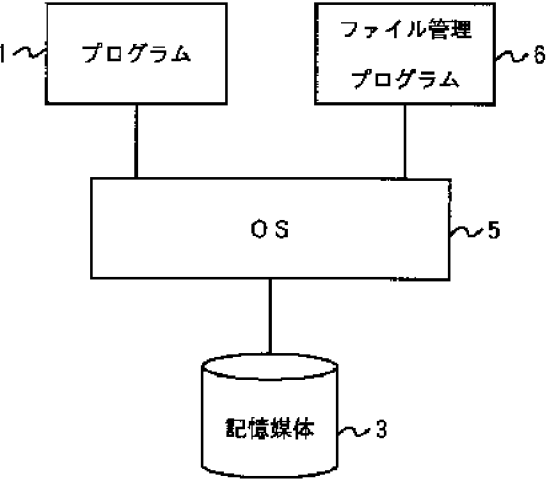
fd=creat ( "/schedule/95.6" ) ; (1)

rval=write ( fd, buf, 256 ) ;      (2)
  
```

[Drawing 11]

/news/616.1	1
/news/616.2	5
/news/616.3	7
/news/616.4	2
/schedule/95.1	10
/schedule/95.2	10
/schedule/95.3	10

[Drawing 12]

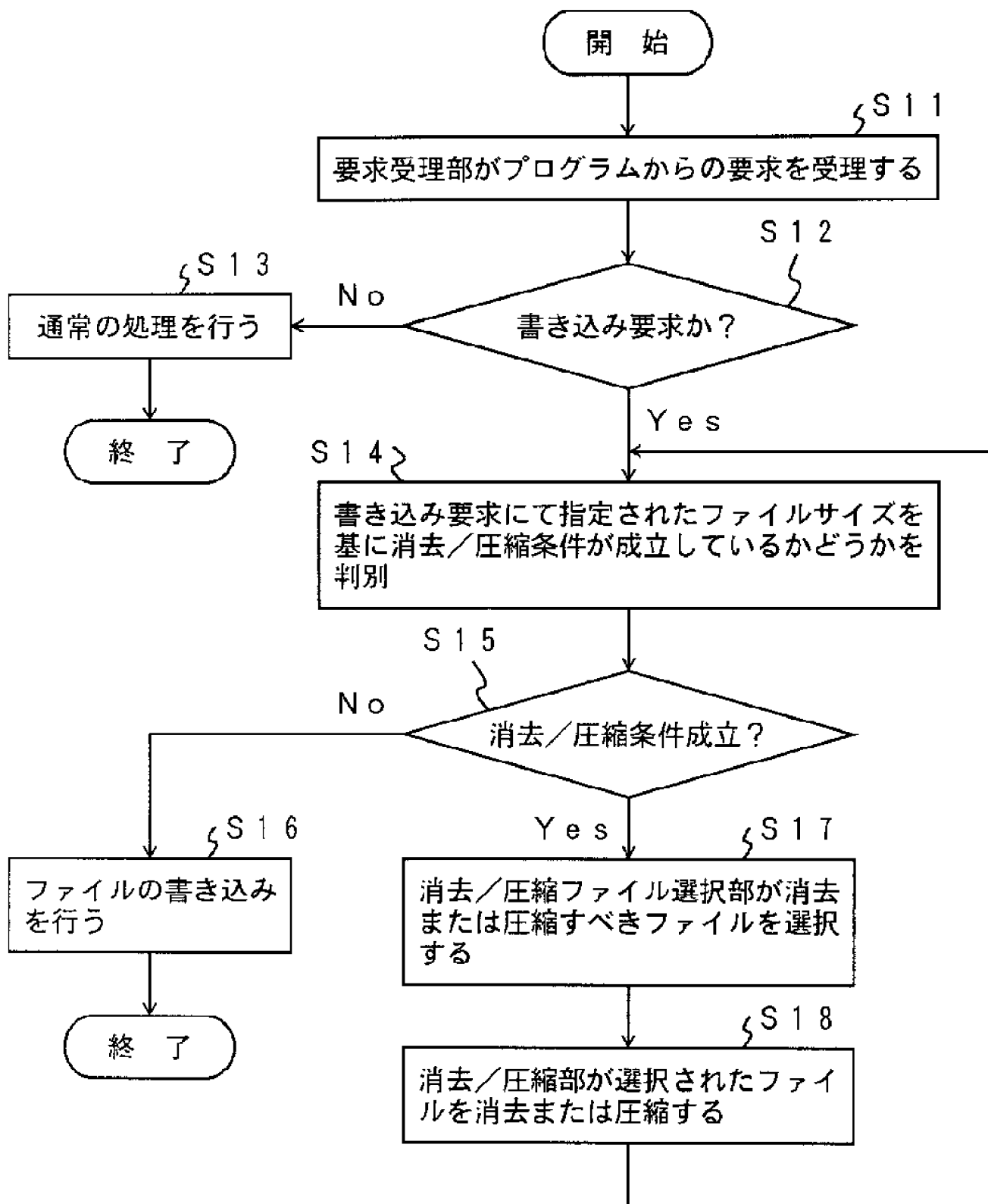


[Drawing 16]

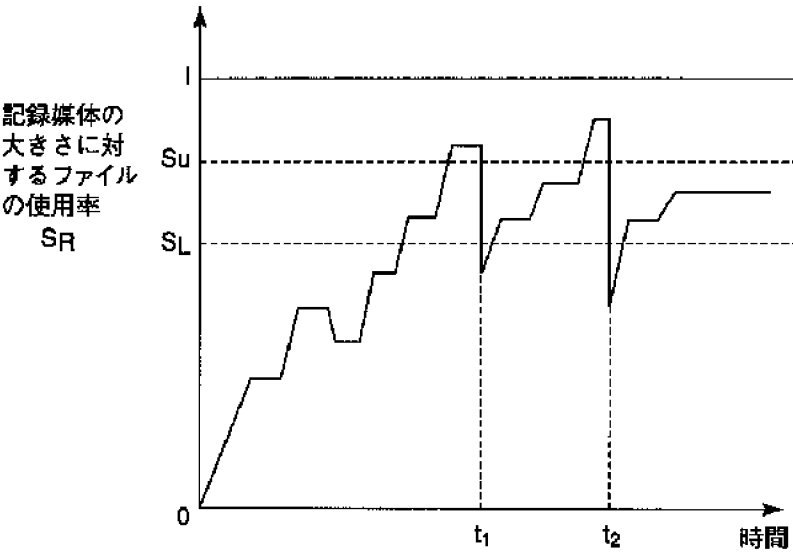
ディレクトリ

ファイル名	ファイル タイプ	----	読出し履歴 (読出し回数)
f 1			1 0
f 2			0
f 3			2 0
f 4			5
⋮	⋮	⋮	⋮

[Drawing 4]



[Drawing 6]



[Drawing 15]

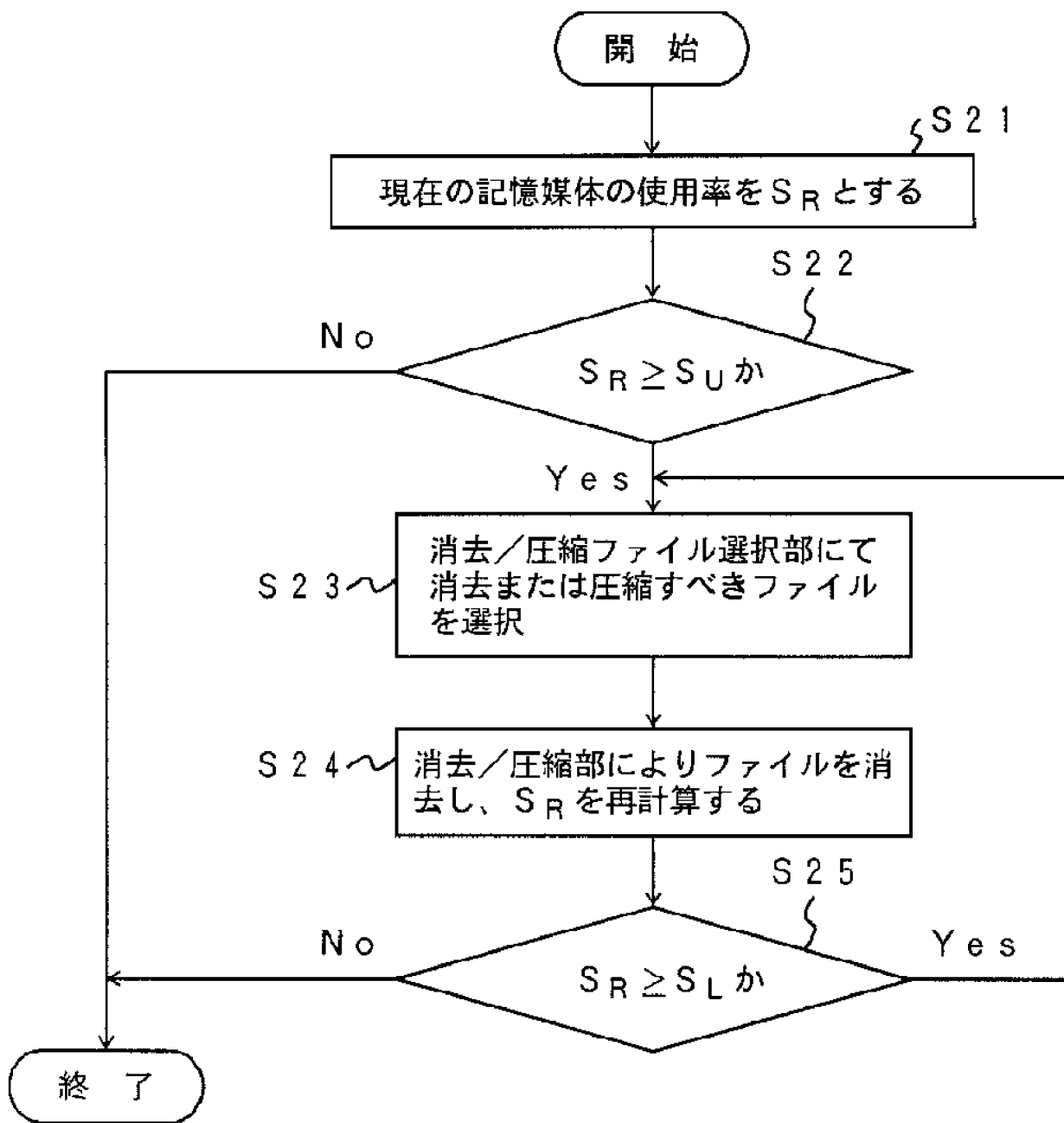
ディレクトリ

ファイル名	ファイル タイプ	----	読出し履歴 (読出しの有無)
f 1			0
f 2			1
f 3			1
f 4			0
⋮	⋮	⋮	⋮

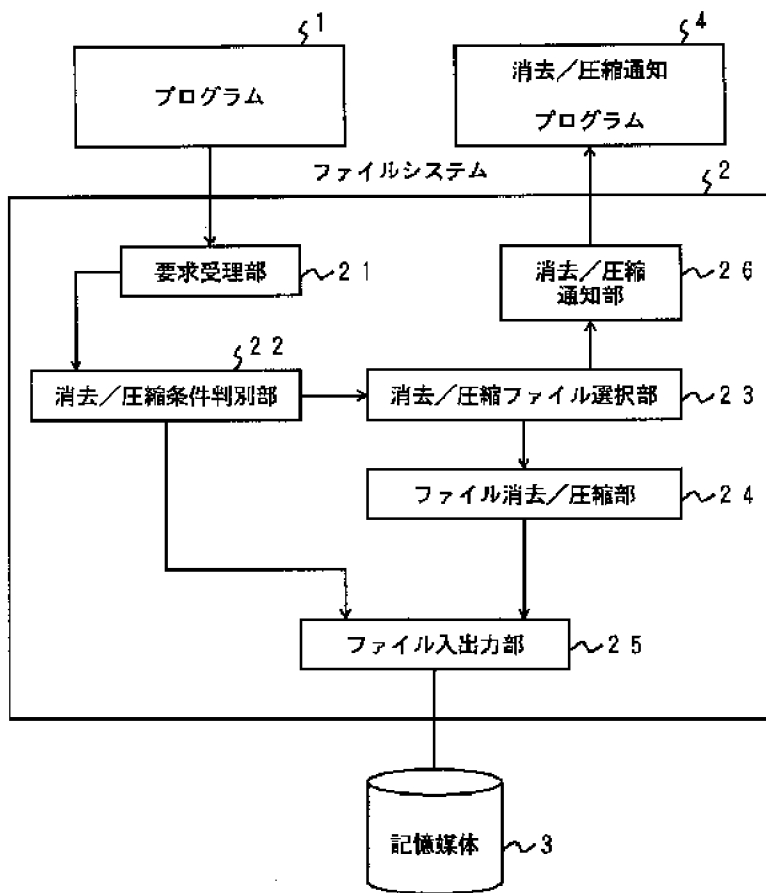
0 : 読出されたことのないファイル

1 : 読出されたことのあるファイル

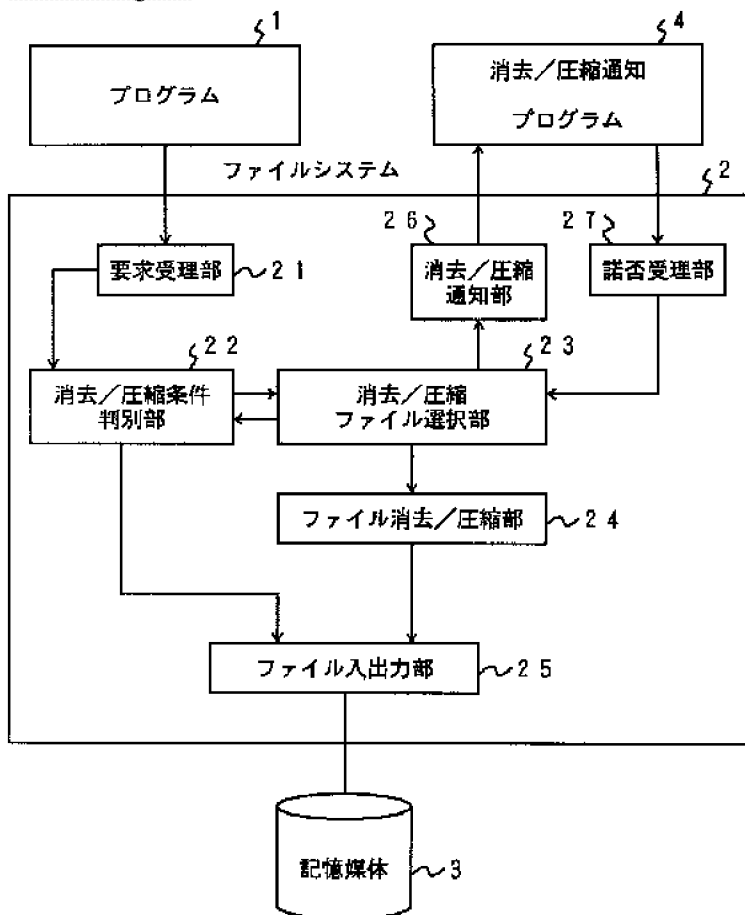
[Drawing 7]



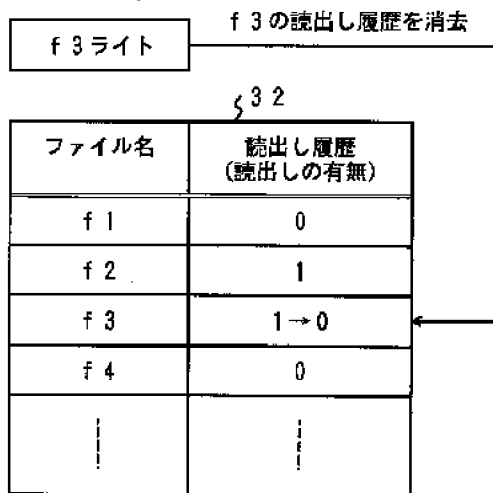
[Drawing 8]



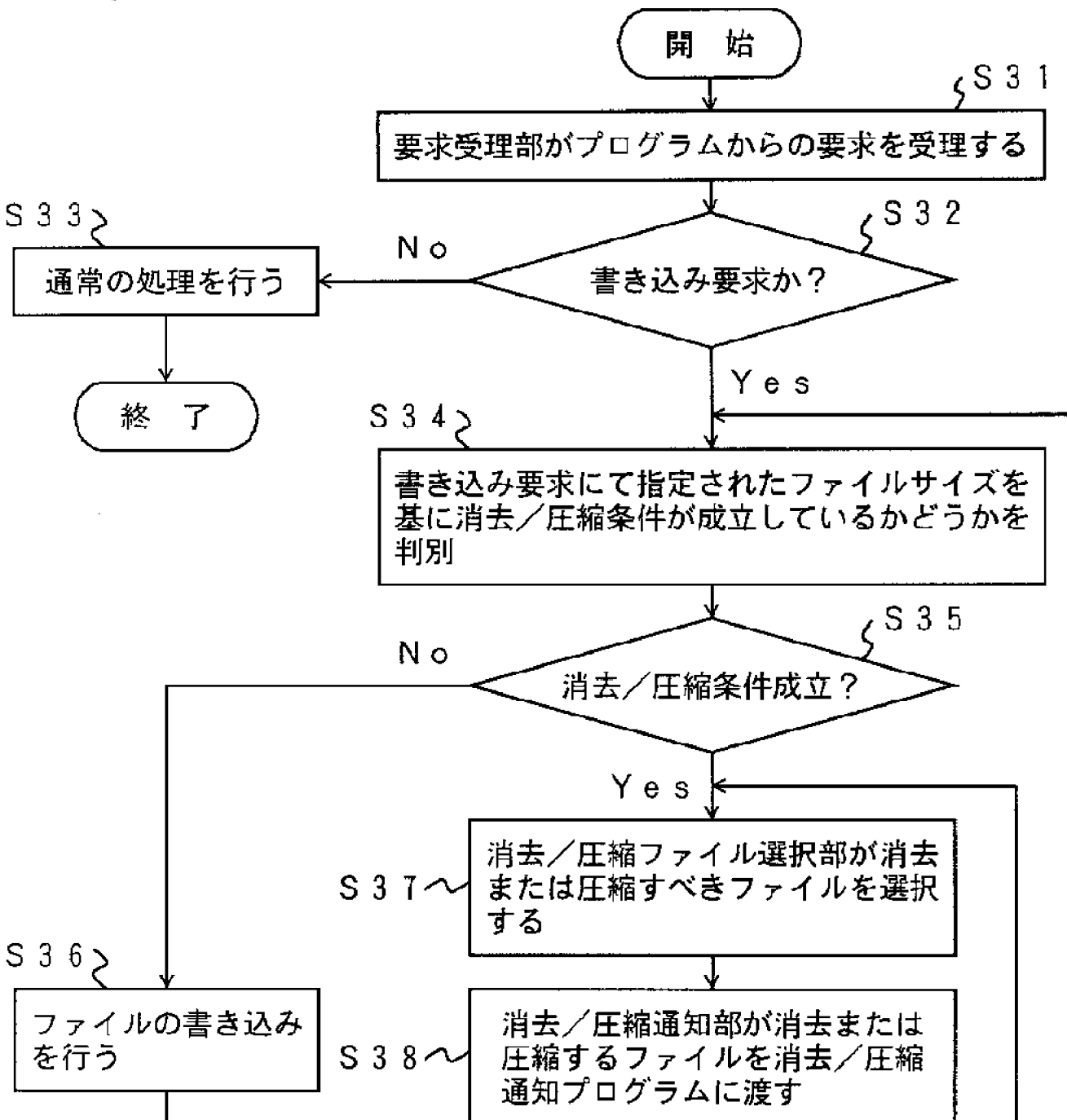
[Drawing 9]

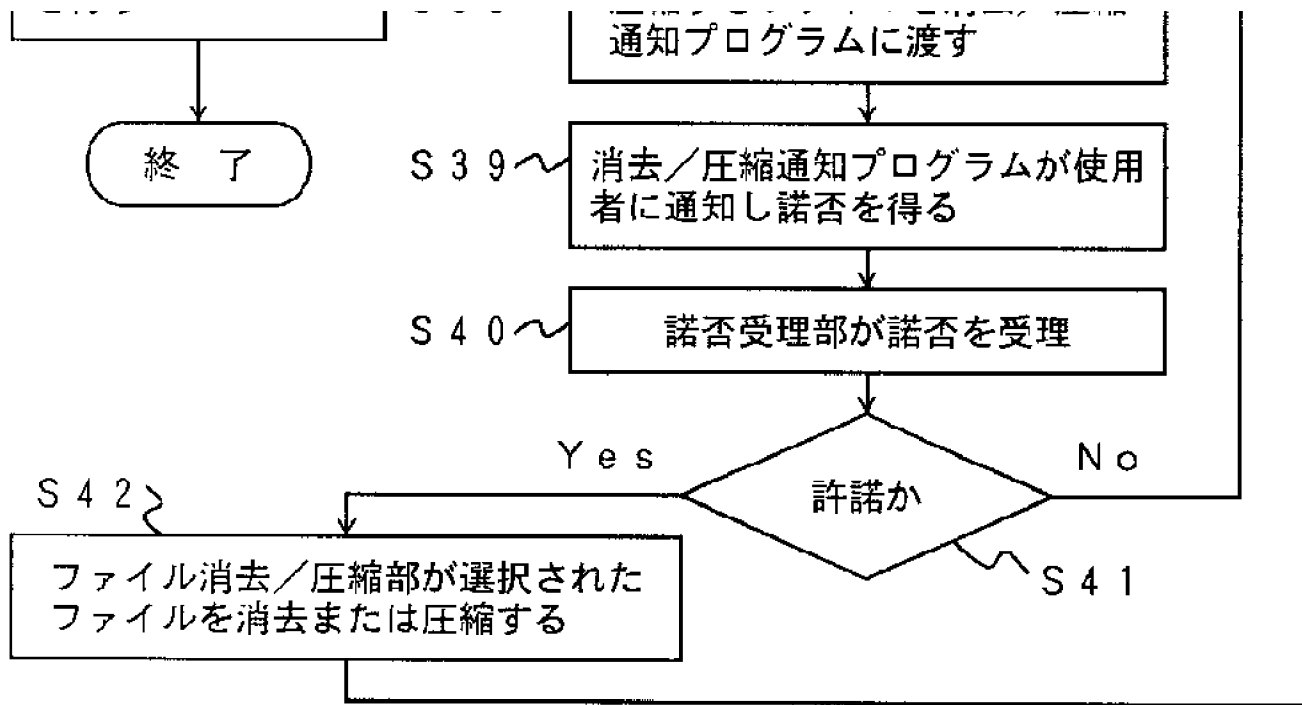


[Drawing 24]



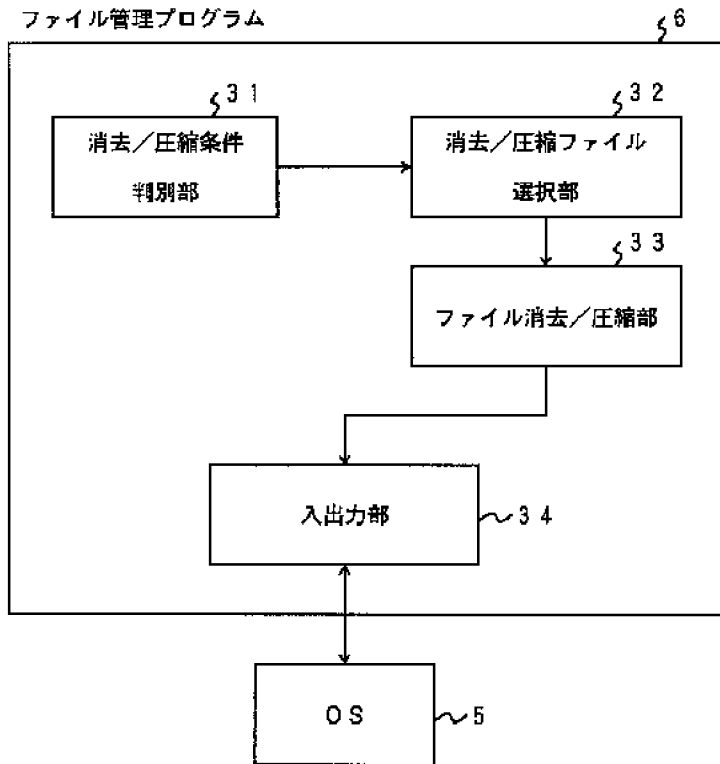
[Drawing 10]



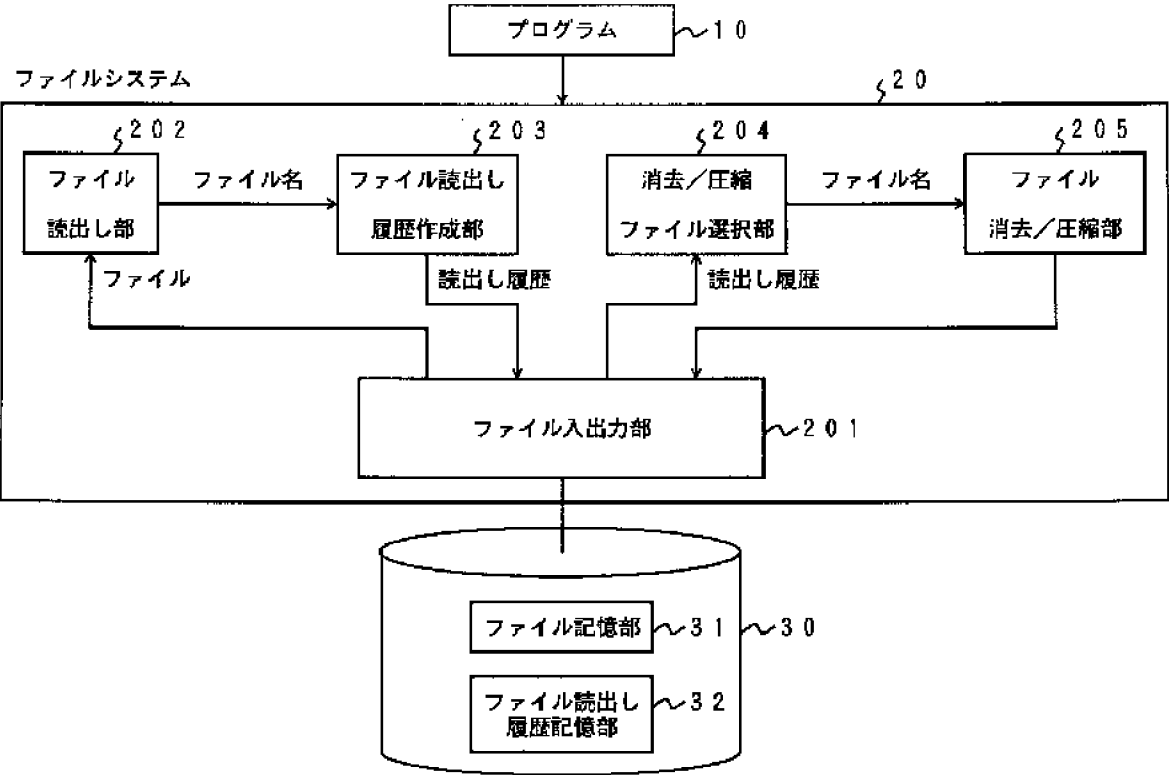


[Drawing 13]

ファイル管理プログラム



[Drawing 14]



[Drawing 17]

ディレクトリ

ファイル名	ファイルタイプ	-----
⋮	⋮	⋮

(A)

読出し履歴ファイル

ファイル名	読出し履歴
⋮	⋮

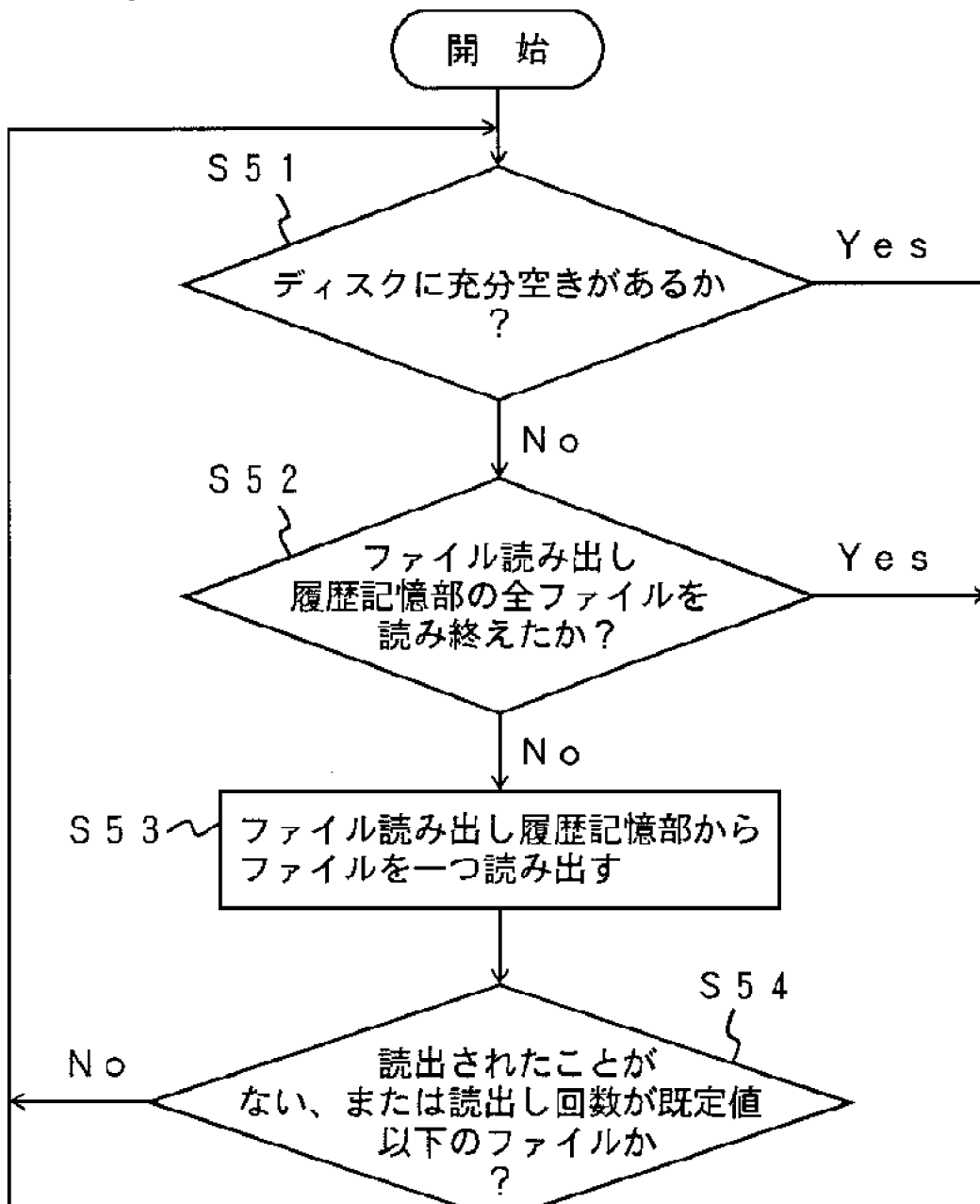
(B)

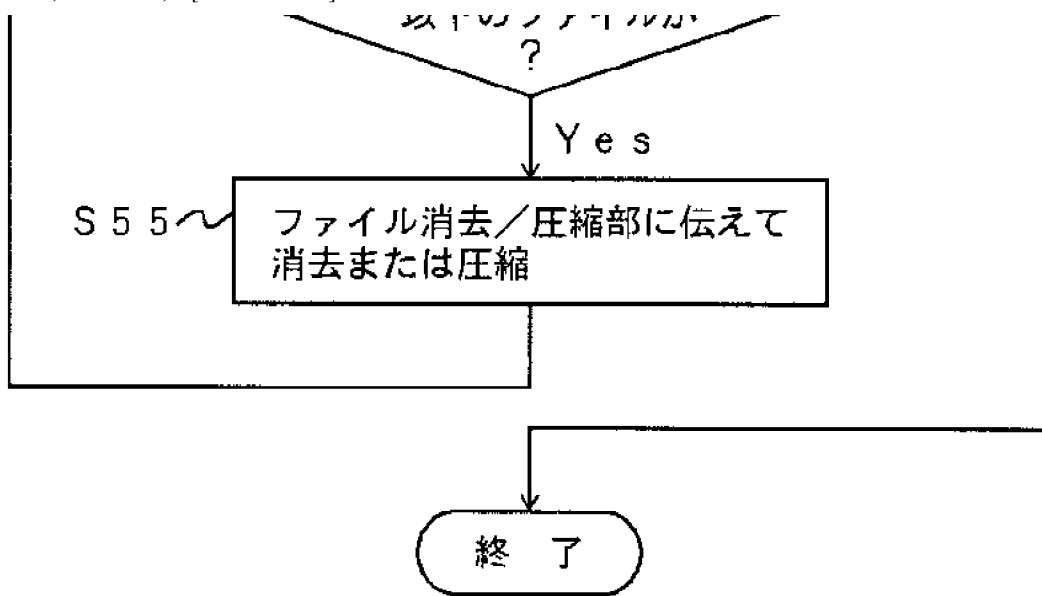
[Drawing 19]

読出し履歴ファイル

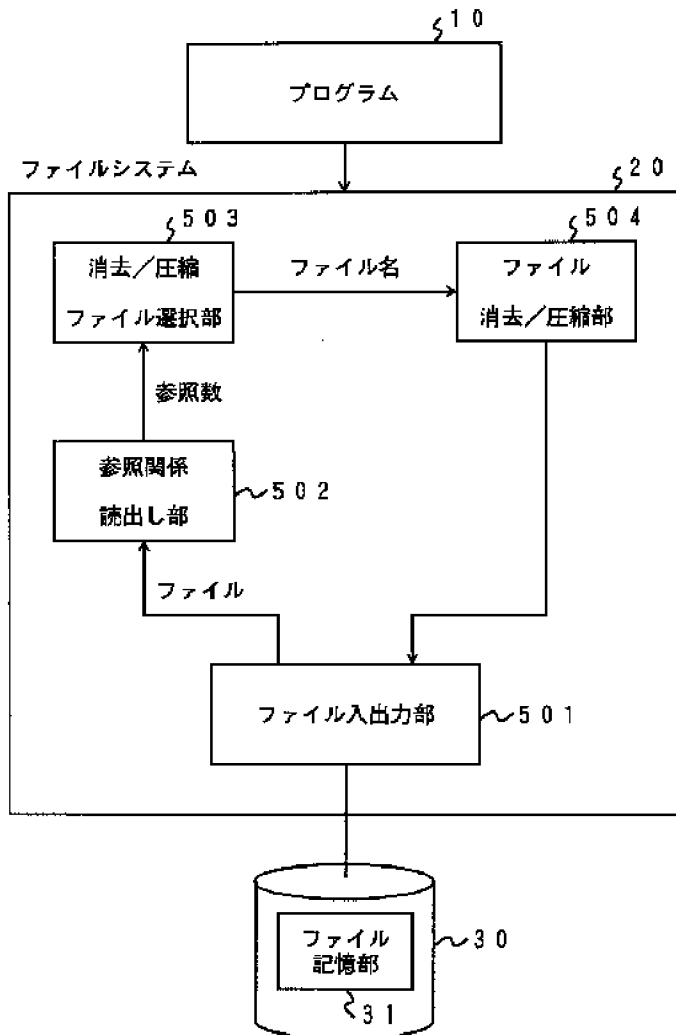
ファイル名	読出し履歴 (最終読出し日時)	
f 1	1 0 : 3 0	9 5 - 8 - 7
f 2	1 7 : 3 0	9 5 - 8 - 6
f 3	1 0 : 3 0	9 5 - 8 - 6
f 4	1 7 : 3 0	9 5 - 8 - 5
f 5	1 0 : 3 0	9 5 - 8 - 5
⋮		

[Drawing 18]

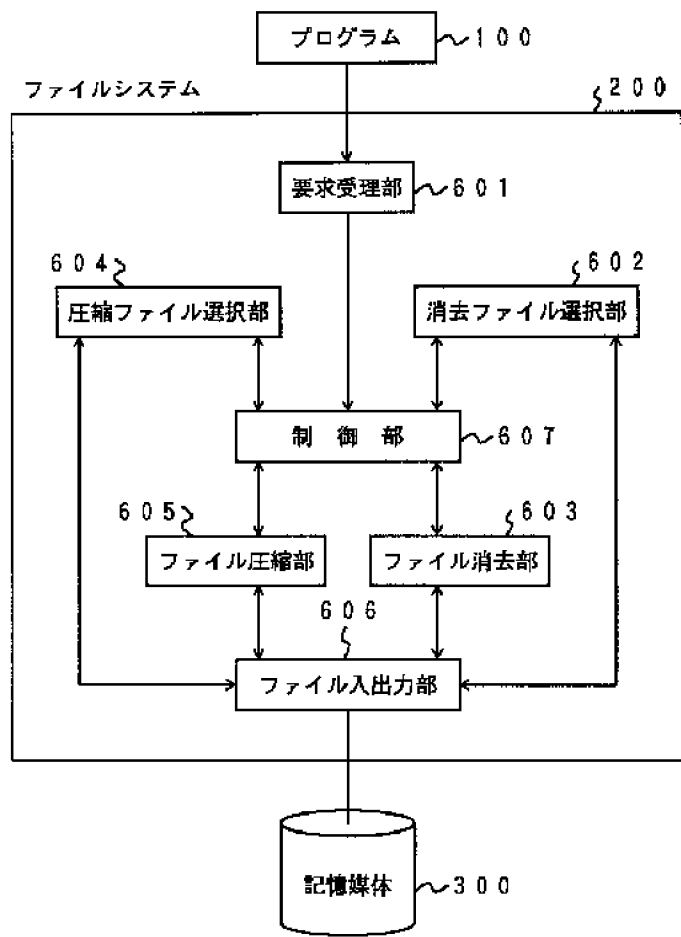




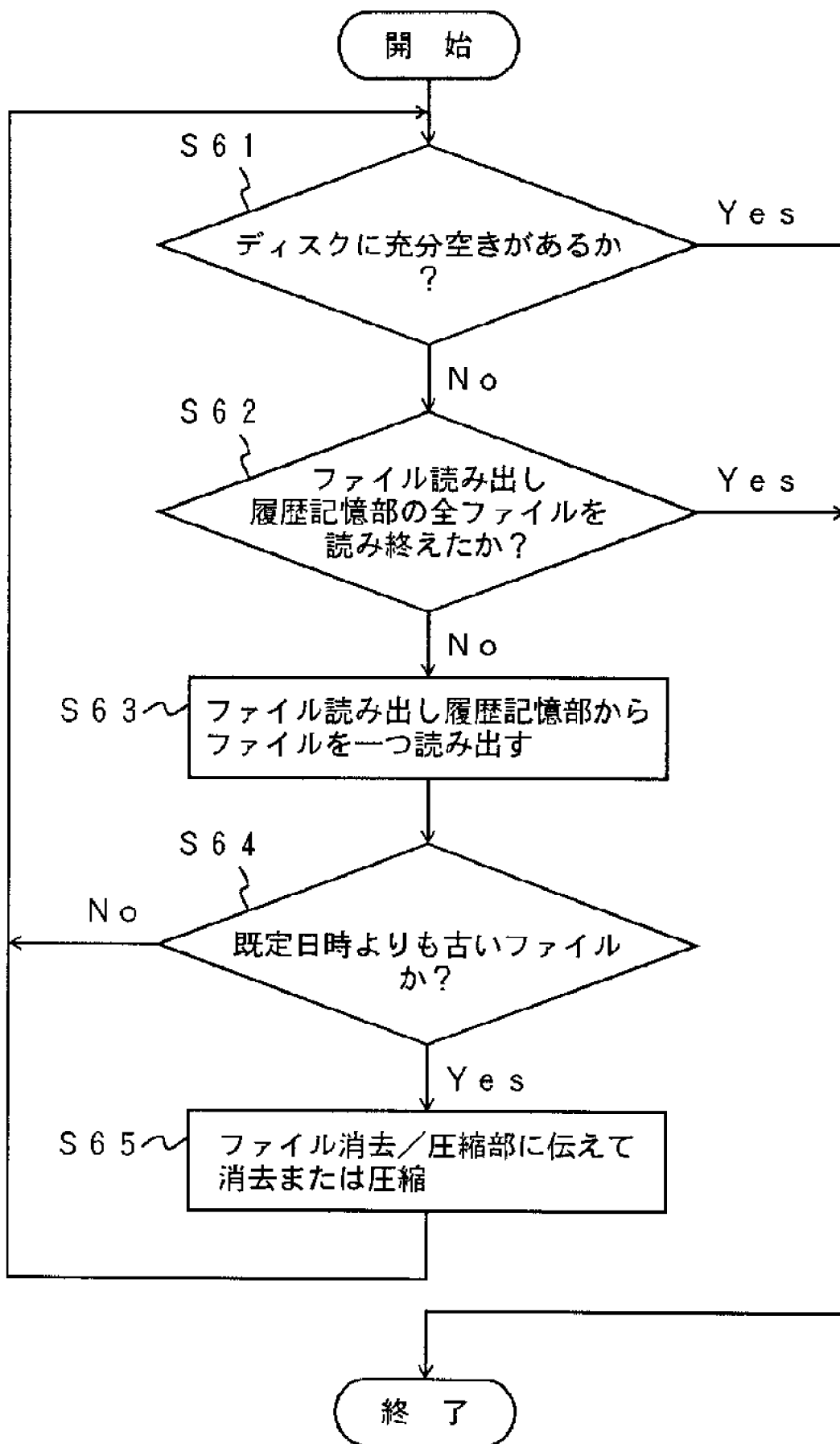
[Drawing 29]



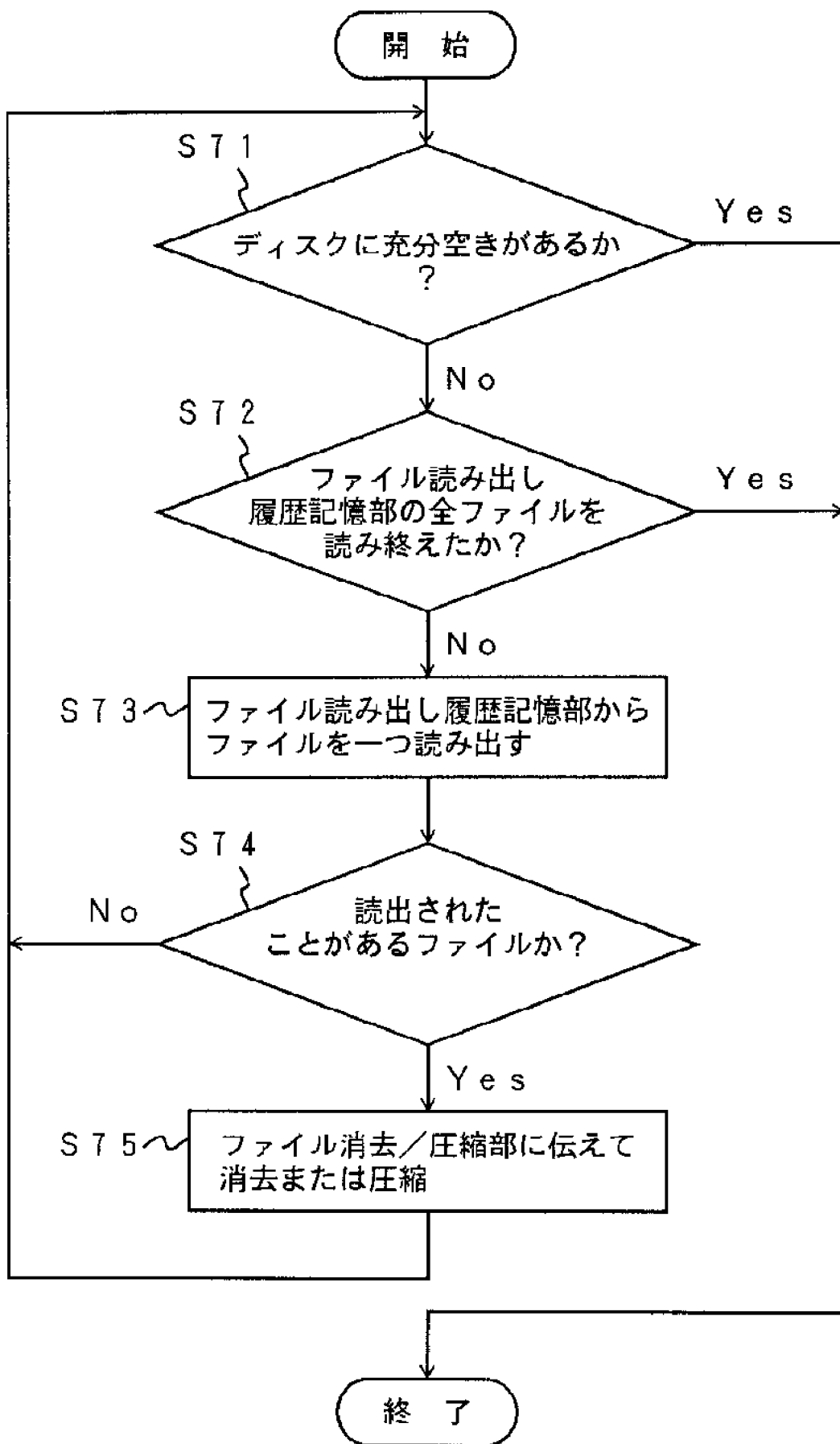
[Drawing 31]



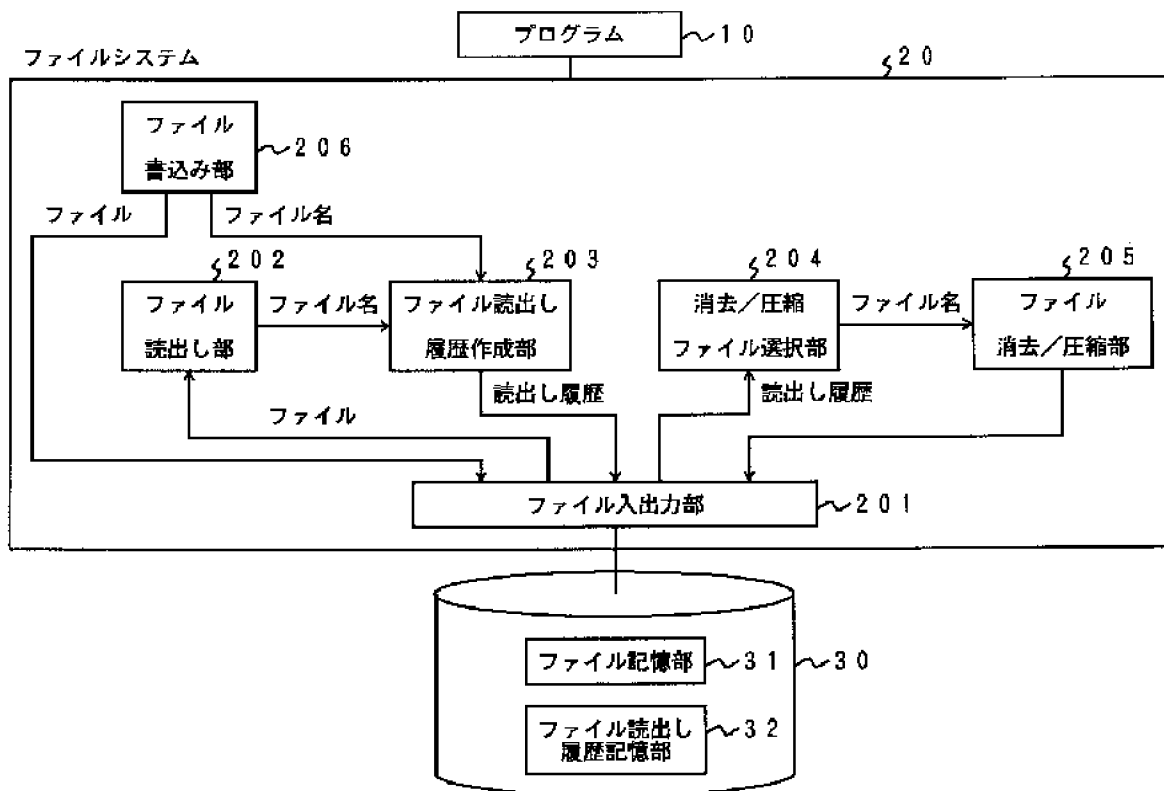
[Drawing 20]



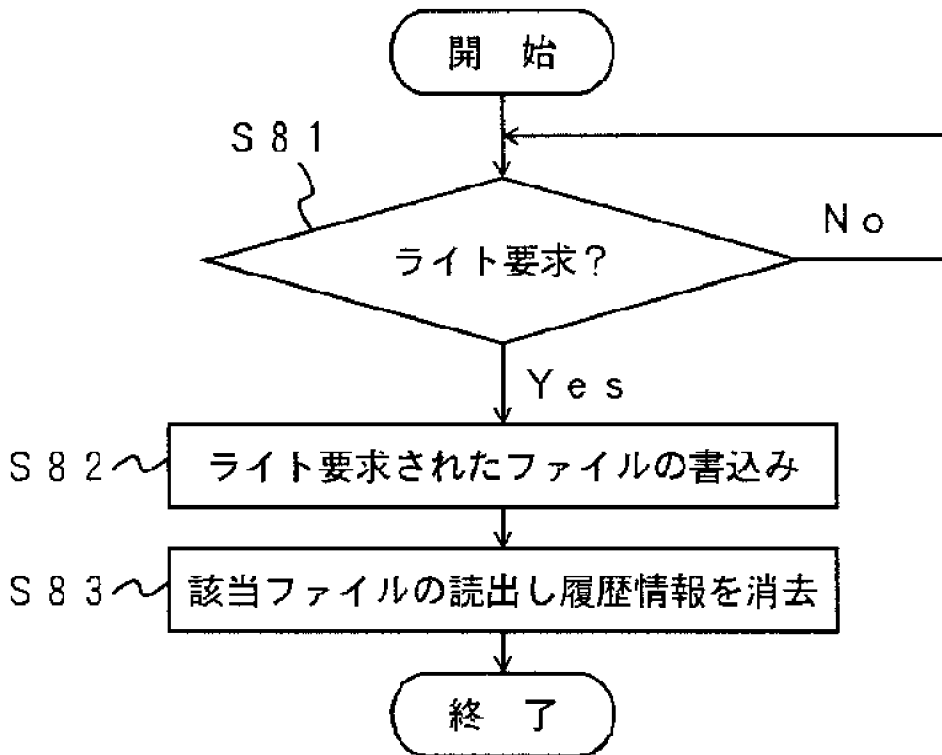
[Drawing 21]



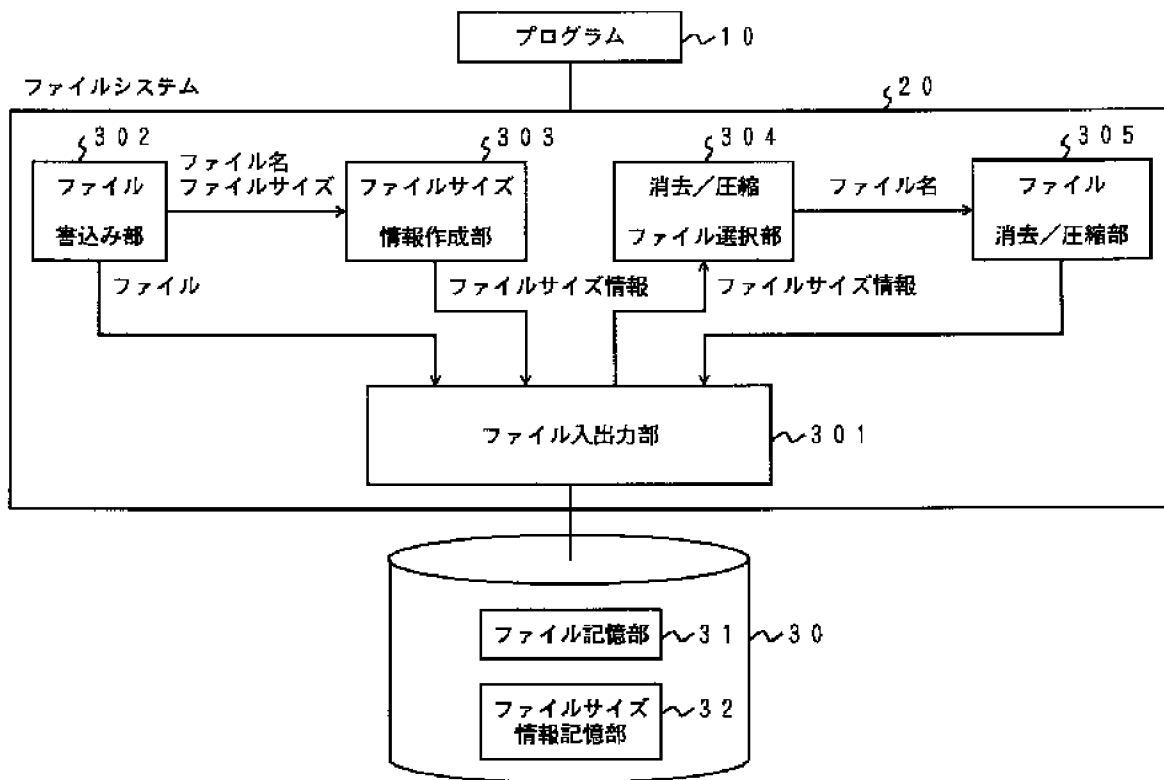
[Drawing 22]



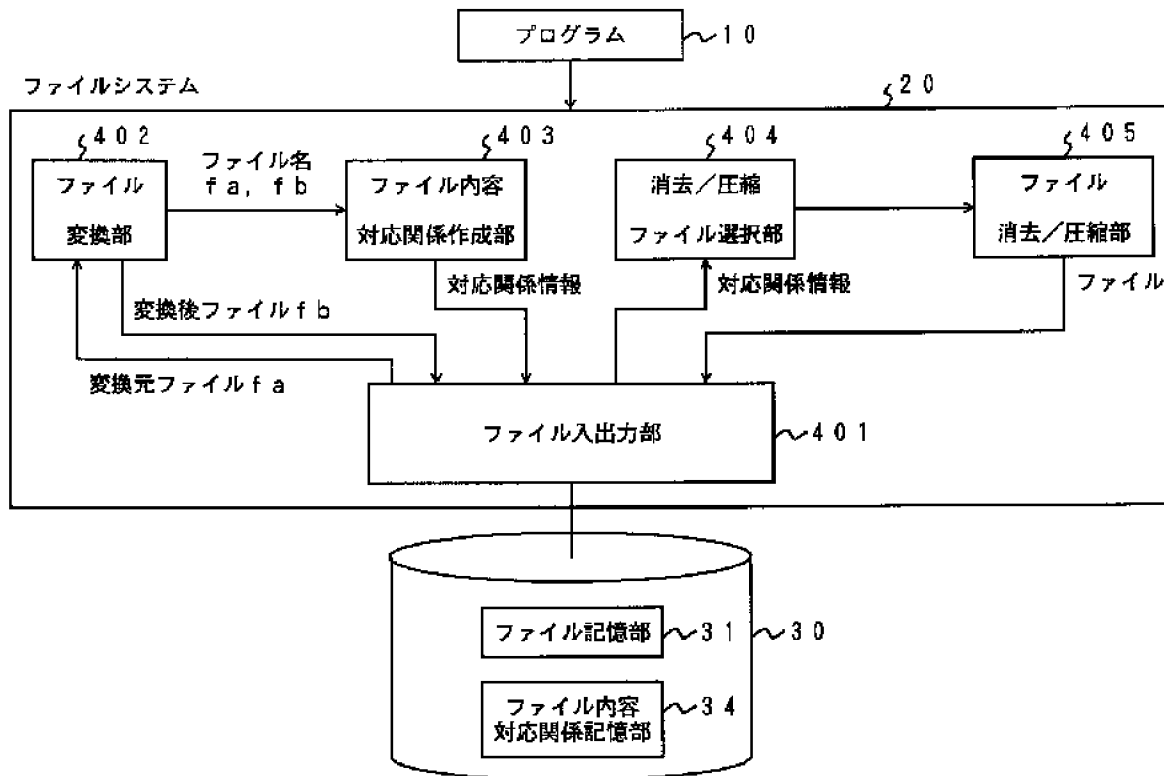
[Drawing 23]



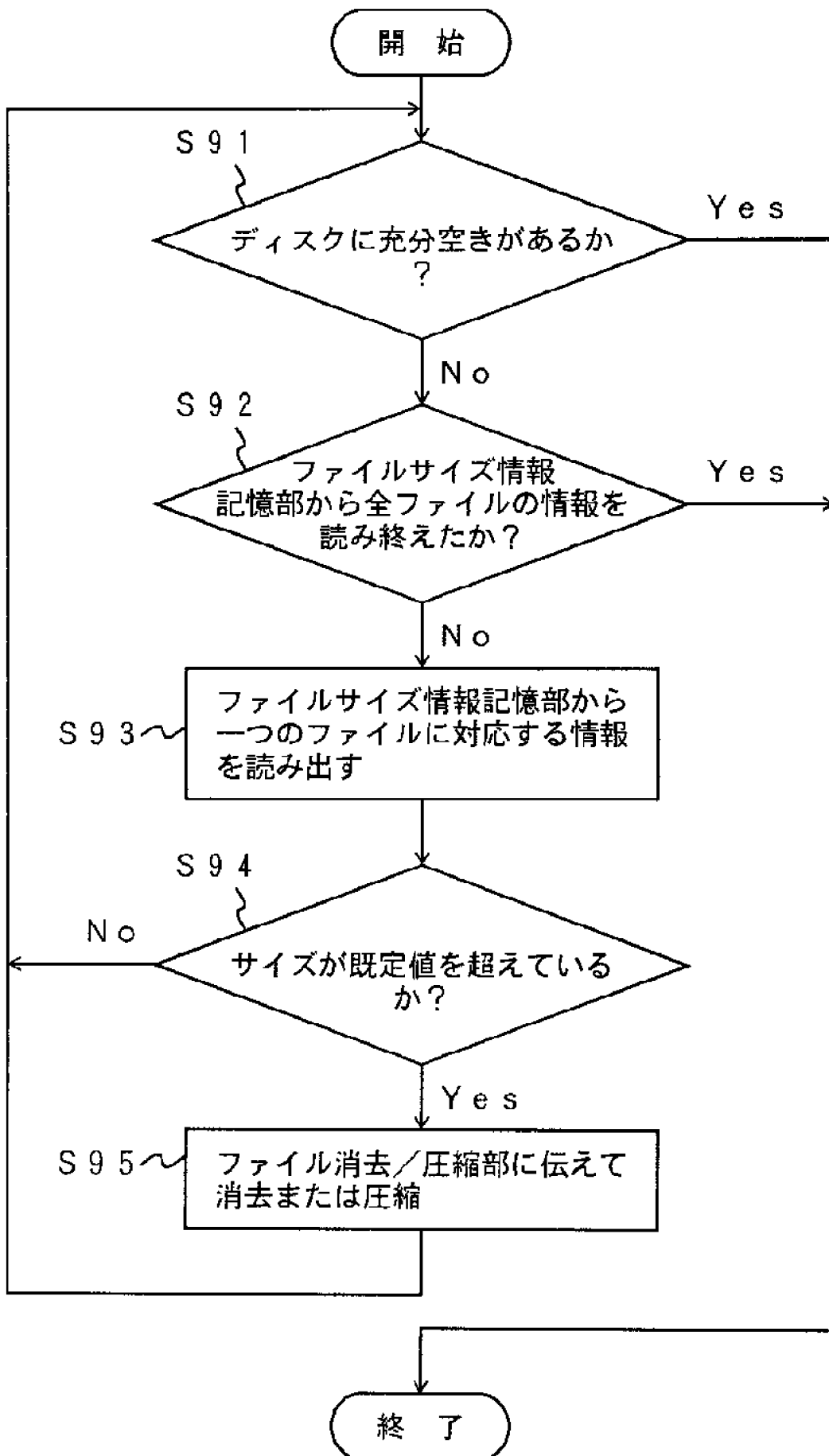
[Drawing 25]



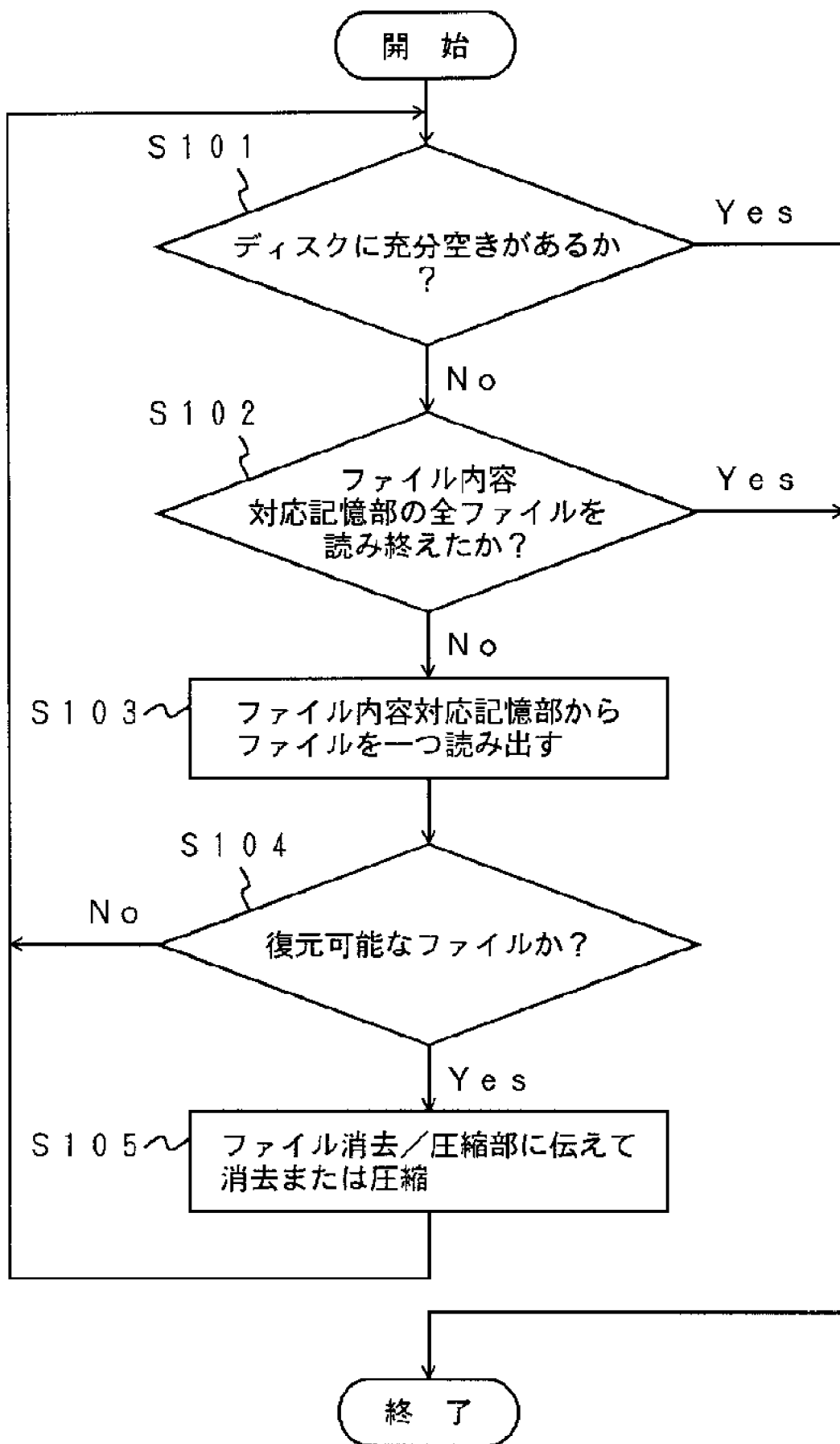
[Drawing 27]



[Drawing 26]



[Drawing 28]



[Drawing 30]

* NOTICES *

JPO and INPIT are not responsible for any damages caused by the use of this translation.

- 1.This document has been translated by computer. So the translation may not reflect the original precisely.
- 2.**** shows the word which can not be translated.
- 3.In the drawings, any words are not translated.

DESCRIPTION OF DRAWINGS

[Brief Description of the Drawings]

[Drawing 1]The block diagram showing the functional constitution of the file system used with the computer system concerning a 1st embodiment of this invention.

[Drawing 2]The figure showing the layered structure of the file treated by the system of the 1st embodiment.

[Drawing 3]The figure showing the internal configuration of the file treated by the system of the 1st embodiment.

[Drawing 4]The flow chart which shows the procedure of the file automatic erasing processing in the system of the 1st embodiment.

[Drawing 5]The figure showing an example of a program which performs a write request to the system of the 1st embodiment.

[Drawing 6]The figure for explaining the erasure condition decision processing used by the file automatic erasing processing in the system of the 1st embodiment.

[Drawing 7]The flow chart which shows the procedure of the erasure condition decision processing performed by the file automatic erasing processing in the system of the 1st embodiment, and elimination file selection processing.

[Drawing 8]The block diagram showing the functional constitution of the file system used with the computer system concerning a 2nd embodiment of this invention.

[Drawing 9]The block diagram showing the functional constitution of the file system used with the computer system concerning a 3rd embodiment of this invention.

[Drawing 10]The flow chart which shows the procedure of the file automatic erasing processing in the system of the 3rd embodiment.

[Drawing 11]The figure showing an example of the importance of the file for choosing an erasing object file in the system of the 3rd embodiment referred to.

[Drawing 12]The figure showing signs that the file system of the 1st embodiment was realized as a user program which became independent of an operating system.

[Drawing 13]The block diagram showing the functional constitution of the file system at the time of realizing the file system of the 1st embodiment as a user program.

[Drawing 14]The block diagram showing the functional constitution of the file system used with the

computer system concerning a 4th embodiment of this invention.

[Drawing 15]The figure showing an example of the file read-out hysteresis information managed by the system of the 4th embodiment.

[Drawing 16]The figure showing other examples of the file read-out hysteresis information managed by the system of the 4th embodiment.

[Drawing 17]The figure showing the example of further others of the file read-out hysteresis information managed by the system of the 4th embodiment.

[Drawing 18]The flow chart which shows the procedure of the file automatic erasing processing in the system of the 4th embodiment.

[Drawing 19]The figure showing another example of the file read-out hysteresis information managed by the system of the 4th embodiment.

[Drawing 20]The flow chart which shows the procedure of other file automatic erasing processings in the system of the 4th embodiment.

[Drawing 21]The flow chart which shows the procedure of other file automatic erasing processings also in the system of the 4th embodiment.

[Drawing 22]The block diagram showing the functional constitution of the file system used with the computer system concerning a 5th embodiment of this invention.

[Drawing 23]The flow chart which shows the procedure of the erasing processing of the file read-out hysteresis information in the system of the 5th embodiment.

[Drawing 24]The figure showing signs that file read-out hysteresis information is eliminated in the system of the 5th embodiment.

[Drawing 25]The block diagram showing the functional constitution of the file system used with the computer system concerning a 6th embodiment of this invention.

[Drawing 26]The flow chart which shows the procedure of the file automatic erasing processing in the system of the 6th embodiment.

[Drawing 27]The block diagram showing the functional constitution of the file system used with the computer system concerning a 7th embodiment of this invention.

[Drawing 28]The flow chart which shows the procedure of the file automatic erasing processing in the system of the 7th embodiment.

[Drawing 29]The block diagram showing the functional constitution of the file system used with the computer system concerning an 8th embodiment of this invention.

[Drawing 30]The flow chart which shows the procedure of the file automatic erasing processing in the system of the 8th embodiment.

[Drawing 31]The block diagram showing the functional constitution of the file system used with the computer system concerning a 9th embodiment of this invention.

[Drawing 32]The figure showing an example of the file management structure used by the system of the 9th embodiment.

[Drawing 33]The flow chart which shows the procedure of the free space secured processing which uses selectively file automatic erasing processing and file automatic compression processing in the system of the 9th embodiment.

[Drawing 34]The flow chart which shows another procedure of the free space secured processing which

uses selectively file automatic erasing processing and file automatic compression processing in the system of the 9th embodiment.

[Description of Notations]

1, 10 -- A user program, 2, 20 -- A file system, 3, 30 -- Storage, 4 -- A notice-of-allowance program, 5 -- An operating system, 6 -- File management program, 21 -- A demand acceptance part, 22 -- Elimination / compression condition discrimination section, 23 -- Elimination / compressed file selecting part, 24 -- File erasure/compression zone, 25 -- A file input output section, 26 -- Elimination/ compression informing part, 27 -- A yes or no acceptance part, 32 -- A file read-out history storage part, 33 -- File size information storage part, 34 -- The relationship storing part corresponding to a file content, 201 -- A file input output section, 202 -- File read section, 203 [-- A file writing part, 303 / -- A file size information preparing part, 403 / -- The relation preparing part corresponding to a file content 502 / -- Reference relation read section.] -- A file read-out history preparing part, 204 -- Elimination / compressed file selecting part, 205 -- A file erasure part, 206

[Translation done.]

(19) 日本国特許庁 (J P)

(12) 公開特許公報 (A)

(11) 特許出願公開番号

特開平9-128276

(43) 公開日 平成9年(1997)5月16日

(51) Int. Cl. ⁶	識別記号	庁内整理番号	F I	技術表示箇所
G 0 6 F 12/00	5 0 1		G 0 6 F 12/00	5 0 1 B 5 0 1 H

審査請求 未請求 請求項の数16 O L (全 33 頁)

(21) 出願番号 特願平8-191845

(22) 出願日 平成8年(1996)7月22日

(31) 優先権主張番号 特願平7-219006

(32) 優先日 平7(1995)8月28日

(33) 優先権主張国 日本 (J P)

(71) 出願人 000003078

株式会社東芝

神奈川県川崎市幸区堀川町72番地

(72) 発明者 今井 敏

神奈川県川崎市幸区小向東芝町1番地 株

式会社東芝研究開発センター内

(72) 発明者 宮田 英樹

神奈川県川崎市幸区小向東芝町1番地 株

式会社東芝研究開発センター内

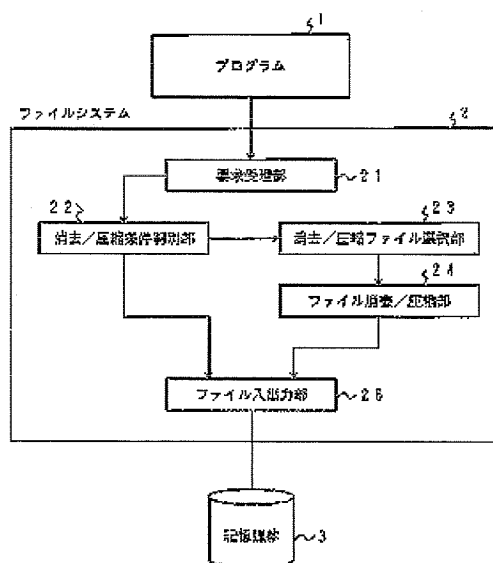
(74) 代理人 弁理士 鈴江 武彦 (外6名)

(54) 【発明の名称】 計算機システムおよびそのシステムで使用するファイル管理方法

(57) 【要約】

【課題】 ファイルを格納するための記憶媒体に自動的に空き領域を生成できるようにし、限られた物理的な記憶媒体の利用効率の向上を図る。

【解決手段】 ファイルシステム2は、記録媒体3の空き領域が所定値以下になった時や、書き込み対象のファイルサイズよりも記録媒体3の空き領域が少ない時に、ファイル自動消去が必要であると判断される。ファイル自動消去が必要であると判断されると、消去ファイル選択部23は、記録媒体3に格納されている各ファイルの優先度にしたがって消去対象のファイルを選択する。そして、その選択されたファイルを消去するための手続きがファイル消去部24によって行われる。従って、ユーザが明示的に消去要求を発生しなくともファイルを格納する記憶媒体に自動的に空き領域を生成できるようになり、物理的な記憶媒体3の限られた記憶容量を効率良く利用することが可能となる。



(2)

特開平9-128276

1

【特許請求の範囲】

【請求項1】 複数のファイルを保存することが可能な記録媒体に対するファイル入出力を管理する計算機システムにおいて、

前記記録媒体の空き領域に基づいて、前記記録媒体のファイルを消去または圧縮する必要があるか否かを判別する手段と、

この判別手段によってファイル消去または圧縮の必要性があることが判定されたとき、前記記録媒体に格納されているファイルの中から消去または圧縮対象のファイルを自動的に選択してそれを消去または圧縮するための手続きを行うファイル自動消去／圧縮手段とを具備することを特徴とする計算機システム。

【請求項2】 前記判別手段の判別処理は、前記計算機システム上で実行されるプログラムからファイル書き込み要求が発行される度、または定期的に実行されることを特徴とする請求項1記載の計算機システム。

【請求項3】 複数のファイルを保存することが可能な記録媒体に対するファイル入出力を管理する計算機システムにおいて、

前記記録媒体の空き領域に基づいて、前記記録媒体のファイルを消去または圧縮する必要があるか否かを判別する手段と、

この判別手段によってファイル消去または圧縮の必要性があることが判定されたとき、前記記録媒体に格納されているファイルの中から、それら各ファイルの優先度に従って消去または圧縮対象のファイルを選択する消去／圧縮ファイル選択手段と、

この消去／圧縮ファイル選択手段によって選択されたファイルを消去または圧縮するための手続きを行うファイル消去／圧縮手段とを具備することを特徴とする計算機システム。

【請求項4】 前記各ファイルの優先度は、ユーザによってファイル毎に予め指定された優先度情報によって規定され、

前記消去／圧縮ファイル選択手段は、前記ユーザによって指定された優先度情報を参照して消去または圧縮対象ファイルを選択することを特徴とする請求項3記載の計算機システム。

【請求項5】 前記各ファイルの優先度は、前記記録媒体に対する各ファイルの入出力履歴情報によって規定され、

前記消去／圧縮ファイル選択手段は、前記入出力履歴情報を参照して消去または圧縮対象ファイルを選択することを特徴とする請求項3記載の計算機システム。

【請求項6】 記録媒体を利用して複数のファイルを保存および管理する計算機システムにおいて、

前記記録媒体に格納されているファイル毎に前記記録媒体からの読み出しの有無、読み出し回数、または最終読み出し日時を示すファイル読み出し履歴情報を保持する

2

手段と、

前記ファイル読み出し履歴情報を参照して、前記記録媒体に格納されているファイルの中から消去または圧縮対象のファイルを選択する消去／圧縮ファイル選択手段と、

この消去／圧縮ファイル選択手段によって選択されたファイルを消去または圧縮するための手続きを行うファイル消去／圧縮手段とを具備することを特徴とする計算機システム。

10 【請求項7】 複数のファイルを保存することが可能な記録媒体に対するファイル入出力を管理する計算機システムにおいて、

前記記録媒体に格納されているファイル毎にファイルサイズを示すファイルサイズ情報を保持する手段と、

前記ファイルサイズ情報を参照して、前記記録媒体に格納されているファイルの中から消去または圧縮対象のファイルを選択する消去／圧縮ファイル選択手段と、

20 この消去／圧縮ファイル選択手段によって選択されたファイルを消去または圧縮するための手続きを行うファイル消去／圧縮手段とを具備することを特徴とする計算機システム。

【請求項8】 複数のファイルを保存することが可能な記録媒体に対するファイル入出力を管理する計算機システムにおいて、

前記記録媒体に格納されているファイルの中で相互に変換可能な内容を持つファイル同士の関係を示すファイル内容対応情報を保持する手段と、

前記ファイル内容対応情報を参照して、前記記録媒体に格納されているファイルの中から他のファイルの内容から復元可能なファイルを消去対象ファイルとして選択する消去ファイル選択手段と、

30 この消去ファイル選択手段によって選択されたファイルを消去するための手続きを行うファイル消去手段とを具備することを特徴とする計算機システム。

【請求項9】 複数のファイルを保存することが可能な記録媒体に対するファイル入出力を管理する計算機システムにおいて、

前記記録媒体に格納されているファイルそれぞれについて他のファイルからの参照数を検出する手段と、

その検出された参照数に従って、前記記録媒体に格納されているファイルの中から消去または圧縮対象のファイルを選択する消去／圧縮ファイル選択手段と、

この消去／圧縮ファイル選択手段によって選択されたファイルを消去または圧縮するための手続きを行うファイル消去／圧縮手段とを具備することを特徴とする計算機システム。

【請求項10】 計算機システムの記録媒体の空き領域を管理するファイル管理方法であって、

前記記録媒体の空き領域に基づいて、前記記録媒体のファイルを消去または圧縮する必要があるか否かを判別

50

(3)

特開平9-128276

3

し、
ファイル消去または圧縮の必要性があることが判別されたとき、前記記録媒体に格納されているファイルの中から消去または圧縮対象のファイルを自動的に選択してそれを消去または圧縮するための手続きを実行し、前記記録媒体に空き領域を自動的に確保できるようにしたことを特徴とするファイル管理方法。

【請求項11】 計算機システム上で実行され、その計算機システムのファイル記憶装置の空き領域を管理するファイル管理プログラムが格納された記録媒体であって、

前記ファイル管理プログラムには、
前記記録媒体の空き領域に基づいて、前記記録媒体のファイルを消去または圧縮する必要性があるか否かを判別し、

ファイル消去または圧縮の必要性があることが判別されたとき、前記記録媒体に格納されているファイルの中から消去または圧縮対象のファイルを自動的に選択し、それを消去または圧縮するための手続きが記述されていることを特徴とする記録媒体。

【請求項12】 複数のファイルを保存することが可能な記録媒体に対するファイル入出力を管理する計算機システムにおいて、

前記記録媒体に格納されているファイルの中から消去対象のファイルを選択してそれを消去するための手続きを行うファイル消去手段と、

前記記録媒体に格納されているファイルの中から圧縮対象のファイルを選択してそれを圧縮するための手続きを行うファイル圧縮手段と、

前記記録媒体の空き領域に基づいて前記記録媒体のファイルを消去または圧縮する必要性があるか否かを判別し、ファイル消去または圧縮の必要性があると判定したとき、前記ファイル消去手段および前記ファイル圧縮手段の一方を選択して実行させる手段とを具備することを特徴とする計算機システム。

【請求項13】 複数のファイルを保存することが可能な記録媒体に対するファイル入出力を管理する計算機システムにおいて、

前記記録媒体に格納されているファイルの中から消去対象のファイルを選択してそれを消去するための手続きを行うファイル消去手段と、

前記記録媒体に格納されているファイルの中から圧縮対象のファイルを選択してそれを圧縮するための手続きを行うファイル圧縮手段と、

前記記録媒体の空き領域に基づいて前記記録媒体のファイルを消去または圧縮する必要性があるか否かを判別する手段と、

この判別手段によってファイル消去または圧縮の必要性があることが判定されたとき、前記記録媒体から圧縮してもよいファイルが無くなるまで、前記ファイル圧縮手

4

段の実行を前記ファイル消去手段よりも優先させる手段とを具備することを特徴とする計算機システム。

【請求項14】 複数のファイルを保存することが可能な記録媒体に対するファイル入出力を管理する計算機システムにおいて、

前記記録媒体の空き領域に基づいて、前記記録媒体のファイルを消去または圧縮する必要性があるか否かを判別する手段と、

この判別手段によってファイル消去または圧縮の必要性があることが判定されたとき、前記記録媒体に格納されているファイルの中から、それら各ファイルの優先度に従って消去対象のファイルを選択する消去ファイル選択手段と、

この消去ファイル選択手段によって選択されたファイルを消去するための手続きを行うファイル消去手段と、

前記判別手段によってファイル消去または圧縮の必要性があることが判定されたとき、前記記録媒体に格納されているファイルの中から、それら各ファイルの優先度に従って圧縮対象のファイルを選択する圧縮ファイル選択手段と、

この圧縮ファイル選択手段によって選択されたファイルを圧縮するための手続きを行うファイル圧縮手段と、

前記選択された消去対象ファイルと圧縮対象ファイルとの関係に基づいて、前記ファイル消去手段と前記ファイル圧縮手段を選択的に実行させる手段とを具備することを特徴とする計算機システム。

【請求項15】 計算機システムの記録媒体の空き領域を管理するファイル管理方法であって、

前記記録媒体の空き領域に基づいて、前記記録媒体のファイルを消去または圧縮する必要性があるか否かを判別し、

ファイル消去または圧縮の必要性があることが判別されたとき、前記記録媒体に格納されているファイルの中に圧縮してもよいファイルが存在するか否かを判別し、

圧縮してもよいファイルが存在しているとき、その圧縮してもよいファイルの中から圧縮対象のファイルを選択してそれを圧縮するための手続きを実行し、

圧縮してもよいファイルが存在しないとき、前記記録媒体に格納されているファイルの中から消去対象のファイルを選択してそれを消去するための手続きを実行することを特徴とするファイル管理方法。

【請求項16】 計算機システムの記録媒体の空き領域を管理するファイル管理方法であって、

前記記録媒体の空き領域に基づいて、前記記録媒体のファイルを消去または圧縮する必要性があるか否かを判別し、

ファイル消去または圧縮の必要性があることが判別されたとき、前記記録媒体に格納されているファイルの中から、それら各ファイルの優先度に従って消去対象ファイルおよび圧縮対象ファイルをそれぞれ選択し、

(4)

特開平9-128276

5

それら選択された消去対象ファイルと圧縮対象ファイルとの関係に基づいて、前記消去対象ファイルの消去または前記圧縮対象ファイルの圧縮を行うための手続きを実行することを特徴とするファイル管理方法。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】この発明は、計算機システムおよびそのシステムで使用されるファイル管理方法に関する。

【0002】

【従来の技術】一般に、計算機システムにて永続的に所有するデータは、ファイルという論理的な単位でディスクのような不揮発性記憶媒体に格納される。物理的な不揮発性記憶媒体の各々の記憶部位にファイルという論理的なデータ単位を割り当てるソフトウェアをファイルシステムと呼ぶ。

【0003】ファイルシステムは物理的な記憶媒体を活用してデータを格納しているため、記憶容量には限界がある。新たにファイルを作成したいが空き領域がなくなったときには、従来では、ユーザ自身が消去すべきファイルを明示的に指定してファイル消去のプログラムを実行することにより空き領域を作る必要があった。プログラムはオペレーティングシステムに対しファイルを指定して消去要求を出し、オペレーティングシステムはこれを受理すると、ファイルシステムに記憶されている該当ファイルを無効化し、空き領域を作っていた。つまり、ファイルシステムに格納するファイルのサイズが増え、空き領域がなくなったり少なくなったりした場合には、ユーザが消去すべきファイルを選び、プログラムを起動することにより明示的にオペレーティングシステムに対して消去要求を行いファイルを消すことにより空き領域を作らなければならなかった。

【0004】

【発明が解決しようとする課題】上述のように、従来では、記憶媒体に十分な空き領域がないときに、ユーザ自身が明示的に消去要求を発生してファイルの消去を行わなければならなかった。

【0005】このため、ユーザが消去すべきファイルを選択するという操作を行ったり、消去するためのプログラムの使用法を覚えなければならず、計算機の使い勝手を悪くする一因となっていた。

【0006】また、ネットワークなどを介して外部からデータが転送されてくる場合には、空き領域がなくなったことに使用者が気がつかないと転送データをファイルシステムに格納することができず、結果としてデータの受信を失敗する原因となっていた。

【0007】この発明はこのような点に鑑みてなされたものであり、ユーザが明示的に消去要求などを発生しなくともファイルを格納する記憶媒体に自動的に必要な空き領域を確保できるようにし、限られた物理的な記憶媒

6

体の記憶容量を効率良く利用することができる計算機システムおよびファイル管理方法を提供することを目的とする。

【0008】

【課題を解決するための手段】この発明は、複数のファイルを保存することが可能な記録媒体に対するファイル入出力を管理する計算機システムにおいて、前記記録媒体の空き領域に基づいて、前記記録媒体のファイルを消去または圧縮する必要があるか否かを判別する手段と、この判別手段によってファイル消去または圧縮の必要があることが判定されたとき、前記記録媒体に格納されているファイルの中から消去または圧縮対象のファイルを自動的に選択してそれを消去または圧縮するための手続きを行うファイル自動消去／圧縮手段とを具備することを特徴とする。

【0009】この計算機システムにおいては、記録媒体の空き領域の大きさに応じてファイルの自動消去または圧縮を行うか否かが判断される。この場合、例えば、記録媒体の空き領域が所定値以下になった時や、書き込み対象のファイルサイズよりも記録媒体の空き領域が少ない時などには、ファイル自動消去または圧縮による空き領域確保処理の実行が必要であると判断される。ファイル自動消去または圧縮による空き領域確保処理が必要であると判断されると、記録媒体に格納されているファイルの中から消去または圧縮対象のファイルが自動的に選択される。そして、その選択されたファイルを消去または圧縮するための手続きが行われる。このファイル消去／圧縮の手続きでは、ファイル自動消去／圧縮手段自らによるファイル消去／圧縮、あるいはオペレーティングシステムに対してファイル消去要求またはファイル圧縮要求を発行することによって、そのオペレーティングシステムによるファイル消去が行われる。従って、ユーザが明示的に消去要求や圧縮要求を発生しなくともファイルを格納する記憶媒体に自動的に空き領域を生成できるようになり、物理的な記憶媒体の限られた記憶容量を効率良く利用することが可能となる。よって、ユーザが消去すべきファイルや圧縮すべきファイルを選択したり、消去または圧縮するためのプログラムの使用法を覚える必要がなく、使い勝手の良い計算機システムを実現できる。また、ネットワークや他の計算機からデータが転送されてくる場合に、ユーザが空き領域を作らなくても転送データを失わず保存することができる。

【0010】ファイル自動消去または圧縮による空き領域確保処理を行うか否かの判別処理は、ユーザプログラムなどからファイル書き込み要求が発行されたとき、あるいは定期的に行うことが好ましい。これにより、常に最適な空き領域を確保することが可能となり、ユーザプログラムによって書き込み指定された新規ファイルが記憶容量不足で保存できなかったり、ネットワークなどからのファイルのダウンロードに失敗するといった不具合

(5)

特開平9-128276

7

を防止できる。

【0011】消去または圧縮対象ファイルの選択は、そのファイル消去またはファイル圧縮によってユーザに及ぼす影響の大きさを考慮してファイルの優先度に従って消去対象ファイル、圧縮対象ファイルを選択することが望ましい。ここで、ファイルの優先度とは、そのファイル消去または圧縮によってユーザにどの程度の影響を及ぼすかを示す指標である。どのファイルが重要であるかはその計算機システムを使用しているユーザ自身の判断によって決まるものである。優先度としては、ユーザによってファイル毎に予め指定された優先度情報や、記録媒体からの各ファイルのファイル読み出しの有無、読み出し回数、最終読み出し日時などに関する入出力履歴情報を使用することが好ましい。

【0012】ユーザによる優先度の指定は、新規ファイルの保存時にそのファイルに付与するファイル名そのものによって行ってもよい。ファイル名とは別の専用のパラメータを優先度としてファイルに付与してもよい。このようなユーザ指定の優先度情報を使用することにより、そのユーザ自身の重要度判断に基づいた消去ファイル選択が可能となる。また、ファイル拡張子を優先度として使用することも出来。この場合には、プログラムファイル、システムファイル、テキストファイル、圧縮ファイル、画像ファイルなどのファイルの種類や、そのファイルを扱うプログラムとの関係などに応じた優先度付けができる。

【0013】また、全てのファイルの優先度が消去ファイルや圧縮ファイルを選択する基準値よりも高く、該当する消去対象ファイルまたは圧縮対象ファイルが存在しない場合には、その基準値を動的に変更したり、それら基準値を越えるファイルの中で最も優先度の低いファイルを消去対象ファイルまたは圧縮対象ファイルとして選択してもよい。

【0014】また、一般には、ユーザによって頻繁に入出力されるファイル程、あるいは最近参照されたファイル程そのユーザにとって重要度が高いファイルと判断できる。一方、ネットワークなどからダウンロードしたニュースや天気予報などのデータファイルなどについては、ユーザがすでに一度参照したことがあるファイルは消しても良いが、まだ参照していないデータは消してはならないという用途も考えられる。したがって、ファイルの入出力履歴、特にファイルの読み出しに関するファイル参照の履歴に基づいて消去ファイルを選択することにより、消去によってユーザに与える影響が大きいファイル、すなわちこれから読み出されることが予想されるファイル、をなるべく消去しないようにすることができる。

【0015】さらに、優先度としてファイルサイズを用い、サイズの大きなファイル程優先的に消去してもよい。これにより、より少ない個数のファイル消去だけで

8

必要な空き容量を確保する事ができる。

【0016】また、この発明の計算機システムは、記録媒体に格納されているファイルの中で相互に変換可能な異なる形式を持つファイル同士（例えば、ファイル形式だけが異なる同一内容のテキストファイルや、データファイルとその圧縮データファイルなど）の関係を示すファイル内容対応情報を保持しておき、そのファイル内容対応情報を参照して復旧可能なファイルを消去対象ファイルとして優先して選択することとを特徴とする。これにより、ユーザに何等影響を及ぼさず、効率的にファイルを消去することができる。なお、この構成は、ファイル消去を行う場合に特に好適であるが、ファイル圧縮を行う場合についても利用することができる。

【0017】また、ハイパーテキストのように他のファイルとの間で参照関係を持つファイルについては、他のファイルからの参照数が多いほどそのファイルを消去または圧縮した時の影響が大きくなるため、記録媒体に格納されているファイルそれぞれについてそのファイルを参照している他のファイルの数などの参照数を検出し、その検出された参照数に従って、前記記録媒体に格納されているファイルの中から消去対象のファイルを選択することが好ましい。

【0018】さらに、この発明の計算機システムは、以上の手段に加え、消去ファイル選択手段により選択されたファイルをユーザに通知して、消去許可を得た上でそのファイルを消去する手段などを適宜組み合わせることで、よりユーザに対する影響の少ない自動ファイル消去を実現できる。

【0019】また、この発明の計算機システムは、ファイル自動消去手段とファイル自動圧縮手段の双方を備え、それらを選択的に使用することとを特徴とする。通常、ファイル圧縮の方がファイル削除を行う場合よりもユーザに及ぼす影響が少ないが、ファイル圧縮よりもファイル削除の方が容易に多くの空き領域を確保することができる。このため、例えば、ユーザによる明示的な指定などによって圧縮と消去を選択的に使用したり、あるいは記録媒体の空き領域サイズなどに基づいて圧縮と消去を自動的に選択すること等により、より効率的なファイル管理が可能となる。

【0020】また、この発明は、ファイル自動消去手段とファイル自動圧縮手段の双方を備え、圧縮してもよいファイルが無くなるまでは（例えば、全てのファイルが圧縮済みであったり、あるいはディレクトリやファイル拡張子などによって予め圧縮してはいけないファイルとして固定的に決められているファイルのようなファイルが残っていないときなど）、ファイル自動圧縮を優先して実行し、圧縮してもよいファイルが無い時にファイル自動圧縮を実行することとを特徴とする。このような圧縮と消去の使い分けにより、ユーザに与える影響をより少なくすることができる。

(6)

特開平9-128276

9

【0021】さらに、圧縮対象ファイルと消去対象ファイルの双方をそれぞれの基準で選択し、選択されたファイルの関係にしたがってファイル消去と圧縮のどちらを行うかを切り替えるようにすることにより、最適な空き領域確保が可能となる。

【0022】

【発明の実施の形態】以下、図面を参照してこの発明の実施形態について説明する。

【0023】図1には、この発明の第1実施形態に係る計算機システムで使用されるファイルシステムの構成が示されている。このファイルシステム2は、記録媒体の空き容量を自動的に確保するための機能として、ファイル自動消去およびファイル自動圧縮の何れか一方を実現するための機能を有している。このファイルシステム2は、オペレーティングシステムの一部として実現されている。このファイルシステム2が適用される計算機システムは、CPU、メモリ、補助記憶装置、および各種I/O装置などのハードウェア資源を持つ通常のコンピュータから構成されるものであるが、ファイルシステム2のファイル自動消去/圧縮機能は、例えば携帯型端末など、ファイル記憶のための補助記憶装置の容量が比較的少ない電子情報機器に好適であるので、以下では、計算機システムがPDAやサブノートパソコンなどの携帯型の電子情報機器である場合を例示して説明する。

【0024】図1に示されているように、この実施形態1の電子情報機器は、その機器のCPUによって実行されるアプリケーションプログラムなどのユーザプログラム1と、そのプログラム1からファイル操作に関する要求を受けてファイルの入出力を管理するファイルシステム2と、ファイルを記憶するための補助記憶装置として使用されるハードディスクやフラッシュメモリカードなどの不揮発性の記憶媒体3とから構成される。

【0025】ファイルに関する操作はユーザプログラム1からだけでなく、ユーザからのキー入力操作に応じて発行されるOSレベルのコマンドによって要求される場合もあるが、どちらの場合でもファイルシステム2から見ると他のプログラムから要求される事になる。したがって、ここでは、ユーザからの要求についてもプログラム1からの要求として扱うことにする。

【0026】ファイルシステム2は、ファイル消去またはファイル圧縮による自動空き領域確保機能の実現のために、図示のように、要求受理部21、消去/圧縮条件判別部22、消去/圧縮ファイル選択部23、ファイル消去部24、およびファイル入出力部25を備えている。

【0027】要求受理部21は、ユーザプログラム1からファイル読み出しや書き込みなどの通常のファイル操作要求を受け付ける。消去/圧縮条件判別部22は、記録媒体3の空き領域に基づいて、記録媒体3のファイルを消去または圧縮して空き領域を確保する必要がある

10

か否か、つまりファイル消去または圧縮によって空き領域確保処理を行う条件（消去/圧縮条件）が満たしているかどうかを判別するためのものであり、要求受理部21にファイル書き込みなどのファイル操作要求が発行される度、あるいはある一定の時間間隔で定期的に起動されて、消去/圧縮条件を満たすか否かの判別処理を行う。

【0028】消去/圧縮ファイル選択部23は、消去/圧縮条件を満たすと判断された時に起動され、記憶媒体3に格納されているファイル（データファイル、プログラムファイルを含む）の中から、それらファイルそれぞれの優先度などに従って消去または圧縮処理の対象となるファイルを選択する。ここで、優先度とは、そのファイルに対して消去または圧縮の処理を施したときにユーザにどの程度の影響を及ぼすかを示す指標であり、及ぼす影響が大きいファイル程、優先度が高いファイルである。

【0029】ファイル消去/圧縮部24は、選択されたファイルに対して消去または圧縮の何れか一方の処理を施すものであり、自動消去機能を有する場合には、選択された消去対象ファイルを記憶媒体3から消去する処理を行い、自動圧縮機能を有する場合には、選択された圧縮対象ファイルを圧縮する処理を行う。ここで、ファイルの消去とはファイル削除と同義であり、オペレーティングシステムによっては用語「ファイル消去」の代わりに「ファイル削除」を使用することもある。

【0030】ファイル消去を行う場合には、消去対象ファイルの実体を記録媒体3から消去（削除）してもよいが、通常のファイルシステムにおけるファイル消去と同様に、そのファイルに関するディレクトリエントリなどの管理データだけを無効化して、ファイルの実体をそのまま残したり、いわゆる「ゴミ箱」と称される専用の記憶領域に移動させるなどの手法を用いてもよい。もちろん、バックアップ用の外部記憶装置を携帯型端末に接続して使用できる場合には、その消去対象ファイルを記録媒体3から消去して外部記憶装置にバックアップファイルとして移してもよい。

【0031】このファイルシステム2で管理されるファイル構造を図2に示す。

【0032】この実施形態では、ファイルを木構造で管理するものとする。すなわち、あるノードの下に複数の子ノードを持ち、各ノードから見た親ノードは一意に定まる。子ノードを持たないノードがファイルでありデータを所有する。子ノードを持つノードはディレクトリと呼ばれる。木構造の最も上位のディレクトリはルートと呼ばれ、親ノードは自分自身である。

【0033】図2においては、“news”ディレクトリの子ノードとしてファイル名「616.1」～「616.4」の4個のファイルが存在し、“schedule”ディレクトリの子ノードとしてファイル名「95.

(7)

特開平9-128276

11

1」～「95. 3」の3個のファイルが存在している様子が示されている。

【0034】図3には、図2に対応する実際のファイル管理構造の一例が示されている。

【0035】図3において、各ノードの第1フィールドは、ノードの種類を表す。Dはディレクトリ、Fはファイルである。第2フィールドは、ファイルまたはディレクトリの名前である。ディレクトリの場合、第3フィールド以降は子ノードへのポインタである。また、図中の斜線は終了を示し、通常はポインタとして使用されない値が登録される。例えばポインタとしてノードの番地を使用し、終了の記しとしてマイナス1を使えば良い。

【0036】ファイルの場合、第3フィールドにそのファイルの重要度を示す値が格納される。この重要度の値は、前述したファイル優先度として用いられ、消去対象ファイルまたは圧縮対象ファイルを選択する時に参照される。図3では、重要度は1から10までの数値で表現され、数値が大きいほど重要度が大きい。この重要度の値は、そのファイルの作成/保存時にユーザによって指定されるものである。また、第4フィールドは実際のデータの大きさを示し、第5フィールドはデータへのポインタである。

【0037】図4には、ファイルシステム2のファイル管理処理の手順を示すアルゴリズムが示されている。

【0038】まず、ファイルシステム2の要求受理部1が、ユーザプログラム1からのファイル操作要求を受け付ける(ステップS11)。そのファイル操作要求がファイル書き込み要求でなければ(ステップS12)、そのファイル操作要求によって指定されたファイル操作(ファイル読み出し、ファイル消去、ディレクトリ参照など)が通常通り行われる(ステップS13)。

【0039】一方、ユーザプログラム1からのファイル操作要求が書き込み要求であるならば(ステップS12)、消去/圧縮条件判別部22は記憶媒体3の空き容量に従って、所定の消去/圧縮条件が成立しているかを判断する(ステップS14)。この場合、書き込み要求されたファイルサイズと記憶媒体3の現在の空き容量とが比較され、空き領域のサイズが書き込み要求されたファイルサイズよりも小さい時に消去/圧縮条件が成立すると判別される。また、書き込み要求されたファイルサイズとは無関係に、記憶媒体3の現在の空き容量が規定値以下であれば消去/圧縮条件が成立すると判別してもよい。

【0040】記憶媒体3の現在の空き容量の値はファイルシステム2内で保持されている。すなわち、ファイルシステム2は、そのファイルシステム2に割り当てられた記憶媒体3の全記憶容量の値をファイル書き込みを行う度にそのファイルサイズ分だけ減算して行き、その結果を記憶媒体3の現在の空き容量として保持している。

【0041】十分な空き容量があって消去または圧縮の

12

条件が成立しないならば(ステップS15)、要求されたファイルの書き込みが直ぐに実行される(ステップS16)。一方、十分な空き容量がなく消去/圧縮条件が成立したならば(ステップS15)、消去/圧縮ファイル選択部23によってファイル選択処理が実行される(ステップS17)。

【0042】すなわち、消去/圧縮ファイル選択部23は、記憶媒体3に格納されているファイルの中で、それぞれファイルに与えられた優先度(図3の重要度)に従って、重要度の値が基準値以下のファイルを選択する。そして、その選択されたファイルは、ファイル消去/圧縮部24によって消去または圧縮される(ステップS18)。

【0043】この後、消去/圧縮条件判別部22によって再び消去/圧縮の条件が成立しているかどうかの判別処理が行われ(ステップS14)、成立しなくなるまでステップS17、S18の処理が繰り返し実行される。そして、消去/圧縮の条件が成立しなくなった時点で、ステップS16のファイル書き込み処理が実行される。

【0044】図5には、要求受理部21にファイル書き込み要求を行う場合のユーザプログラム1の例が示されている。

【0045】まず、プログラム1にて新規ファイルの生成(ファイルクリエイイト)が要求される。この場合、図5の(1)のようにファイル名(「/schedule/95.6」)が指定されて、新たなファイル生成のためのシステムコールがファイルシステム2に送られる。ファイル生成が成功すると、ファイルシステム2からプログラム1にファイル識別子(fd)が返される。つぎに、プログラム1によってファイルへの書き込み要求が行われる。(2)は、ファイル識別子により指定したファイルfdに対して、変数bufで特定されるプログラム中の位置から始まる256バイトを書き出す要求である。つまり、書き込み要求は、書き込み対象のファイルと書き込みバイト数を指定して行われる。要求受理部21はこの要求を受理し、書き込み対象のファイルと、書き込みバイト数を得る。

【0046】次に、消去/圧縮条件判別部22があらかじめ定められた消去/圧縮条件が成立しているかどうかを判別する。この場合、前述したように、記憶媒体3の空き領域が書き込み要求されたファイルサイズを下回る時、あるいは空き領域があらかじめ定められた大きさを下回っているときに消去/圧縮条件が成立したと判断される。

【0047】図6には、消去/圧縮条件判別部22による後者の判別方式の具体例が示されている。

【0048】この判別処理では、ファイルシステム2による記憶媒体3の使用率が一定値SUを越えている時に書き込み要求を受理した場合に別の一定値SLを下回るまでファイル消去または圧縮の空き容量確保処理が繰り返

(8)

特開平9-128276

13

返される。記憶媒体3の使用率はファイル書き込みを行えば増え、消去や部分的な削除、または圧縮を行えば減る。図6においては、時刻t1、t2にて、書き込み要求を受理した際にSUを越えているため、消去を行いSLまで減らした様子が表示されている。

【0049】図7に、図6の消去判別処理に対応するアルゴリズムを示す。

【0050】消去／圧縮条件判別部22は、記憶媒体3の現在の使用率SRを計算し（ステップS21）、これがSU以上であるか否かを調べる（ステップS22）。SR \geq SUであれば、消去／圧縮ファイル選択部23が実行されて、消去または圧縮すべきファイルが選択される（ステップS23）。次いで、その選択されたファイルがファイル消去／圧縮部24にて消去または圧縮された後、再び使用率SRが消去／圧縮条件判別部22によって計算される（ステップS24）。そして、SR \geq SLであるか否かが消去／圧縮条件判別部22によって調べられ、SRがSLを下回るまでステップS23、S24が繰り返される。一般にSL \leq SUである。

【0051】なお、消去／圧縮判別部22が消去／圧縮条件が成立していないと判別した場合には、書き込み要求は通常通りに処理される。すなわち、書き込み権限などをチェックの上、問題がなければ書き込みを行う。

【0052】消去／圧縮判別部22が消去／圧縮条件が成立していると判別した場合、消去／圧縮ファイル選択部23が消去または圧縮すべきファイルを選択するが、この選択のためにはいくつかのアルゴリズムが考えられる。

【0053】例えばファイルシステムに存在するファイルのうち最も重要度の低いファイルから一つを選択する。重要度が最低のファイルが複数ある場合には、任意の一つを選んでも良いし、そのなかで最もファイルサイズの大きいファイルを選んでも良いし、ファイルサイズ最小のファイルを選んでも良い。あるいは書き込み要求されているファイルサイズを下回らない範囲内で、最小ファイルサイズのファイルを選んでも良い。

【0054】特定のファイルについては消去または圧縮の対象にしない場合には、消去／圧縮ファイル選択部23はある値を越える重要度のファイルは消去または圧縮対象としないことにしておき、消去または圧縮対象にしないファイルには設定した値を越える重要度を与えればよい。例えば重要度9を越えるファイルは消去や圧縮の対象としないものとする。図3に示すファイルのうち、「95.1」、「95.2」、「95.3」は消去や圧縮の対象とされない。これにより、特定のファイルは消去または圧縮対象から除くことができる。なお、この値は固定的なものでなく、変更できることにしても良い。

【0055】消去／圧縮条件判別部22により消去または圧縮が必要と判別された場合でも、消去／圧縮ファイ

14

ル選択部23にて消去または圧縮対象ファイルが選択できないこともある。この場合には、例えば次のように対処することが好ましい。

【0056】（1）消去／圧縮ファイル選択部23は、該当するファイルがないことをプログラム1に返したり使用者に通知し、ファイルの消去や圧縮はしない。

【0057】（2）消去／圧縮ファイル選択部23が、あらかじめ設定した値を越える重要度のファイルは選択できないという方式をとっている場合に、ファイルが選択できないならば、その設定値を変更して低くする。あるいはあらかじめ設定した値を越える重要度のファイルの中から最も重要度の低いファイルを選ぶ。これらの場合には、ユーザにその旨を通知し、許諾を得てから行うことが好ましい。

【0058】図8には、この発明の第2の実施形態に係る電子情報機器のシステム構成が示されている。

【0059】ここでは、ファイルシステム2に消去／圧縮通知部26が追加されており、これによって消去または圧縮対象として選択されたファイルがユーザに通知されるように構成されている。

【0060】すなわち、消去／圧縮ファイル選択部23にて選択されたファイルは消去／圧縮通知部26により消去／圧縮通知プログラム4に渡されて、ユーザに通知される。これにより、ユーザは、ファイルが自動消去または圧縮されること、及びどのファイルが消去または圧縮されたかを知ることができる。消去／圧縮通知部26による通知のタイミングは、ファイル消去／圧縮部24によるファイル消去または圧縮の前でも、後でも構わない。

【0061】図9には、この発明の第3の実施形態に係る電子情報機器のシステム構成が示されている。

【0062】ここでは、実施形態2の構成に加え、ファイルシステム2に諾否受理部27が設けられている。

【0063】すなわち、消去／圧縮ファイル選択部23にて選択されたファイルは消去／圧縮通知部26により消去／圧縮通知プログラム4に渡され、使用者に通知される。使用者は消去／圧縮通知プログラム4に対して消去または圧縮を認めるか認めないかを入力する。この諾否は諾否受理部27により受理され、消去／圧縮ファイル選択部24に入力される。もし使用者がそのファイルの消去または圧縮による空き領域確保処理の実行を認めれば、このファイルは消去または圧縮される。認めなければ消去／圧縮ファイル選択部23は別のファイルを選択する。これにより、重要度の低いファイルといえども使用者の知らない間に消去や圧縮がなされることはなく、使用者の許可を受けた上で消去や圧縮が行われる。図10に図9に対応するフローチャートを示す。

【0064】まず、ファイルシステム2の要求受理部1が、ユーザプログラム1からのファイル操作要求を受け付ける（ステップS31）。そのファイル操作要求がフ

50

(9)

特開平9-128276

15

ファイル書き込み要求でなければ(ステップS32)、そのファイル操作要求によって指定されたファイル操作(ファイル読み出し、ファイル消去、ディレクトリ参照など)が通常通り行われる(ステップS33)。

【0065】一方、ユーザプログラム1からのファイル操作要求が書き込み要求であるならば(ステップS32)、消去/圧縮条件判別部22は記憶媒体3の空き容量に従って、所定の消去/圧縮条件が成立しているかを判断する(ステップS34)。この場合、前述したように、書き込み要求されたファイルサイズと記憶媒体3の現在の空き容量とが比較され、空き領域のサイズが書き込み要求されたファイルサイズよりも小さい時に消去/圧縮条件が成立すると判断される。また、書き込み要求されたファイルサイズとは無関係に、記憶媒体3の現在の空き容量が規定値以下であれば消去/圧縮条件が成立すると判断してもよい。空き領域としては、前述した使用率SRを利用することができる。

【0066】十分な空き容量があって消去/圧縮条件が成立しないならば(ステップS35)、要求されたファイルの書き込みが直ぐに実行される(ステップS36)。一方、十分な空き容量がなく消去/圧縮条件が成立したならば(ステップS35)、消去/圧縮ファイル選択部23による消去または圧縮対象ファイルの選択処理が実行される(ステップS37)。すなわち、消去/圧縮ファイル選択部23は、記憶媒体3に格納されているファイルそれぞれについて順番に優先度(図3の重要度)を調べ、重要度の値が基準値以下のファイルを選択する。

【0067】そして、その選択されたファイルは、消去/圧縮通知部26により消去/圧縮通知プログラム4に渡され、使用者に通知される(ステップS38、S39)。使用者は消去/圧縮通知プログラム4に対してそのファイルに消去または圧縮の処理を施すことを認めるか認めないかを入力する。この可否は可否受理部27により受理され(ステップS40)、消去/圧縮ファイル選択部24に入力される。もし使用者がファイル消去やファイル圧縮を認めれば(ステップS41)、このファイルはファイル消去/圧縮部42にて消去または圧縮される(ステップS42)。認めなければ、再び、ステップS37～S41が行われ、消去/圧縮ファイル選択部23は別のファイルを選択する。

【0068】図11には、各ファイルの優先度管理の他の例が示されている。

【0069】すなわち、図3では各ファイルの中に重要度のフィールドを設け、その中にファイルの重要度を示す値を格納したが、ファイルの名前と重要度の対からなる表を図11のように作成し、それを別個のファイルなどとして格納してもよい。この表は、ファイルの生成、削除及び重要度の変更に伴って更新される。

【0070】また、新規ファイルの作成/保存時にその

16

ファイルにユーザが付与するファイル名そのものによって優先度を管理しても良い。また、ファイル拡張子を優先度として使用することも出来、この場合には、プログラムファイル、システムファイル、テキストファイル、圧縮ファイル、画像ファイルなどのファイルの種類や、そのファイルを扱うプログラムとの関係などに応じた優先度付けが可能となる。

【0071】以上、ファイルシステム2がオペレーティングシステムにファイル管理部として組み込まれている場合を想定して実施形態1～3を説明したが、ファイルシステム2のファイル自動消去/圧縮機能はオペレーティングシステムの外でユーザレベルで動くプログラムとして実現することもできる。図12にその場合の構成例を示す。

【0072】すなわち、図12においては、実施形態1～3で説明したファイルシステム2のファイル自動消去/圧縮機能がオペレーティングシステム外で動作するファイル管理プログラム6として実装されており、一般のユーザプログラム1と別個に動作し、直接一般のプログラム1からの要求を受理しない。オペレーティングシステム5から見ると、ファイル管理プログラム6は一般のプログラム1と同様のユーザプログラムに見える。

【0073】図13に、ファイル管理プログラム6の機能構成を示す。

【0074】図13における消去/圧縮条件判別部31、消去/圧縮ファイル選択部32、ファイル消去/圧縮部33は、それぞれ図1の消去/圧縮条件判別部22、消去/圧縮ファイル選択部23、ファイル消去/圧縮部24に相当するが、図13の入出力部34はオペレーティングシステム5とのインタフェースとして機能する。

【0075】消去/圧縮条件判別部31により消去/圧縮条件が成立していると判断されると、消去/圧縮ファイル選択部32が実行されて、消去または圧縮すべきファイルが選択される。選択されたファイルを消去または圧縮すべく、ファイル消去/圧縮部33が消去要求と圧縮要求の何れかを発生する。これは入出力部34を介して、システムコールとしてオペレーティングシステム5に対して送られる。オペレーティングシステム5はこれを受理し、ファイルを消去あるいは圧縮する。消去/圧縮条件判別部31および消去/圧縮ファイル選択部32によって実行される具体的な処理内容は、図6、図7で説明した処理と同じである。

【0076】以上説明したように、実施形態1～3の電子情報機器で使用されるファイルシステム2(ファイル管理プログラム6を含む)によれば、それにファイル自動消去機能あるいはファイル自動圧縮機能の何れか一方を設けることにより、ユーザが明示的に消去要求や圧縮要求を発生しなくともファイルを格納する記憶媒体3に自動的に空き領域を生成できるようになり、物理的な記

(10)

特開平9-128276

17

記憶媒体3の限られた記憶容量を効率良く利用することが可能となる。よって、空き領域が少ない場合でも、ユーザが消去すべきファイルを選択したり、消去するためのプログラムの使用法を覚える必要がなく、使い勝手の良い計算機システムを実現できる。また、ネットワークや他の計算機からデータが転送されてくる場合に、ユーザが空き領域を作らなくても転送データを失わず保存することができる。

【0077】なお、以上の説明では、ユーザプログラム1からファイル書き込み要求が発行されたときに消去／圧縮条件の成立の有無を判定したが、この判定処理はある時間間隔で定期的に行ってもよい。これは、ユーザプログラム1と独立している図13の構成に好適である。

【0078】次に、図14乃至図21を参照して、この発明の第4の実施形態を説明する。

【0079】前述したように実施形態1～3ではファイルごとに固定的に決められた重要度の値を消去／圧縮ファイル選定のための優先度として使用したが、実施形態4では、各ファイルの入出力に関する履歴、すなわちユーザによるファイルの使用状況を、消去／圧縮ファイル選定のための優先度として使用する構成である。このよう

な優先度を使用するのは、以下の理由によるものである。

【0080】すなわち、通常、計算機システムでは、頻繁に参照されるファイルとあまり頻繁に参照されないファイルとが存在する。頻繁に参照されるファイルを消してしまうと、ユーザが再びそのファイルを参照しようとしてもファイルが見つからない、という問題が生じる可能性が高い。同様に、頻繁に参照されるファイルを圧縮してしまうと、そのファイルの読み出しの度に伸張処理が必要とされ、ファイル読み出しに要する時間が長くなるという問題が発生する。

【0081】また、これとは逆に、ネットワークなどからダウンロードしたニュースなどの文書ファイルについては一旦読み出されたことがあるものはユーザにとって重要度が低いファイルであると想定する事ができるので、ユーザがすでに一度参照したファイルは消してもよいが、まだ参照していないファイルは消してはならない、という用途もある。

【0082】したがって、実施形態4では、このような状況に対応するために、ファイルの読み出しに関する履歴情報が優先度として使用される。以下、実施形態4の機能構成を説明する。

【0083】この実施形態4の電子情報機器は、前述した実施形態1～3のシステム構成と同様に、アプリケーションプログラムなどのユーザプログラム10と、そのプログラム10からファイル操作に関する要求を受けてファイルの入出力を管理するファイルシステム20と、ファイルを記憶するための補助記憶装置として使用されるハードディスクやフラッシュメモリカードなどの不揮

18

発性の記憶媒体30とから構成される。

【0084】ファイルシステム20は、ファイル入出力部201、ファイル読み出し部202、ファイル読み出し履歴作成部203、消去／圧縮ファイル選択部204、およびファイル消去／圧縮部205から構成されている。

【0085】ファイルの読み出し時には、ファイル読み出し部202はファイル入出力部201を介して記憶媒体30のファイル記憶部31からファイルを読み出すほか、ファイル読み出し履歴作成部203に該当ファイルのファイル名とそのファイルの読み出しがあった旨を通知する。ファイル読み出し履歴作成部203は、ファイル読み出し部202からの通知に従って、ファイル読み出しの有無、読み出し回数、最終読み出し日時などをファイル毎に示すファイル読み出し履歴情報を作成する。このファイル読み出し履歴情報は、ファイル入出力部201を介して記憶媒体30のファイル読み出し履歴記憶部32に格納される。

【0086】消去／圧縮ファイル選択部204は、ファイル入出力部201を介してファイル読み出し履歴記憶部32のファイル読み出し履歴情報を参照し、ファイル記憶部31に格納されているファイルの中からユーザにとって重要でないと想定されるファイルを消去または圧縮対象ファイルとして選択し、そのファイル名をファイル消去／圧縮部205に通知する。この消去／圧縮ファイル選択部204の消去／圧縮ファイル選択処理は、実施形態1～3と同様に、プログラム10からファイル書き込みなどのファイル操作要求が発行された時、あるいは定期的に行われる。

【0087】ファイル消去／圧縮部205は、選択されたファイルを消去または圧縮する。この場合、ファイルの消去や圧縮は、実施形態1～3と同様の方式で行われる。

【0088】次に、図15～図17を参照して、ファイル読み出し履歴情報を具体的に説明する。

【0089】図15の例では、ファイル読み出し履歴情報はディレクトリ内にファイル管理情報の1つとして保持されている。ファイル読み出し履歴情報は、“0”または“1”のフラグによって表され、フラグ“0”は読み出されたことがないファイルであることを示し、フラグ“1”は読み出されたことがあるファイルであることを示す。

【0090】図16の例では、ファイル読み出し履歴情報として、ファイル読み出しの有無を示すフラグではなく、ファイル読み出し回数を示す値が利用されており、それが図15と同様に各対応するファイルのディレクトリエントリに保持されている。

【0091】これら図15または図16のファイル読み出し履歴情報は、図17に示されているように、ディレクトリとは別個に、読み出し履歴ファイルとして保存す

(11)

特開平9-128276

19

ることもできる。ファイルの最終読み出し日時についても、ファイル読み出しの有無や読み出し回数と同様に管理することができる。また、オペレーティングシステムによってはそれ自体にファイル毎の最終読み出し日時を管理する機能を持つものがあるので、それを利用してもよい。

【0092】次に、図18を参照して、ファイル読み出し履歴を利用したファイル自動消去／圧縮動作を説明する。ここでは、参照頻度の低いファイルを優先して削除あるいは圧縮する場合を想定する。

【0093】まず、実施形態1～3と同じ手法で記憶媒体30のファイル記憶部31に十分な空き領域があるか否か、すなわち消去／圧縮条件が成立しているか否かが調べられる（ステップS51）。消去／圧縮条件が成立したならば、消去／圧縮ファイル選択部204の処理が開始される。すなわち、まず、消去／圧縮ファイル選択部204は、ファイル読み出し履歴記憶部32から1つのファイルに関する読み出し履歴情報を読み出し（ステップS53）、そのファイルが読み出されたことがない、あるいは読み出し回数が所定値以下のファイルであるか否かを調べる（ステップS54）。読み出されたことがない、あるいは読み出し回数が所定値以下のファイルであれば、消去／圧縮ファイル選択部204は、そのファイル名をファイル消去／圧縮部205に伝えて消去または圧縮させる（ステップS55）。

【0094】この後、再度、消去／圧縮条件が成立するか否かが調べられ（ステップS51）、消去／圧縮条件が成立しなくなるか、あるいは全てのファイルについての読み出し履歴情報の読み出しが終了するまで、ステップS53～S55が繰り返される。

【0095】このように、読み出し履歴を用いて消去や圧縮を行うファイルを決断することによって、頻繁に読み出され、これからも読み出されることが予想されるファイルなるべく消去あるいは圧縮しないようにすることができる。よって、ファイル消去／圧縮によってユーザに与える影響を大幅に低減する事ができる。

【0096】図19には、ファイル読み出し履歴情報の他の例が示されている。

【0097】ここでは、ファイル読み出し履歴情報として各ファイルの最終読み出し日時情報が用いられている。この場合、ファイルの読み出し時には、ファイル読み出し部202はファイル記憶部31からファイルを読み出すほか、ファイル読み出し履歴作成部203に該当ファイルのファイル名を通知する。ファイル読み出し履歴作成部203は、通知されたファイル名と現在の日時とを対応させてファイル読み出し履歴記憶部32に記憶する。

【0098】次に、図20を参照して、ファイル読み出し履歴を利用したファイル自動消去／圧縮動作を説明する。ここでは、基本的には、最終読み出し日時が古いフ

20

ァイル程優先して削除または圧縮される。

【0099】まず、実施形態1～3と同じ手法で記憶媒体30のファイル記憶部31に十分な空き領域があるか否か、すなわち消去／圧縮条件が成立しているか否かが調べられる（ステップS61）。消去／圧縮条件が成立したならば、消去／圧縮ファイル選択部204の処理が開始される。すなわち、まず、消去／圧縮ファイル選択部204は、ファイル読み出し履歴記憶部32から1つのファイルに関する読み出し履歴情報を読み出し（ステップS63）、最終読み出し日時が予め決められた所定の日時よりも古いファイルであるか否かを調べる（ステップS64）。最終読み出し日時が所定の日時よりも古いファイルであれば、消去／圧縮ファイル選択部204は、そのファイル名をファイル消去／圧縮部205に伝えて消去あるいは圧縮させる（ステップS65）。

【0100】この後、再度、消去／圧縮条件が成立するか否かが調べられ（ステップS61）、消去／圧縮条件が成立しなくなるか、あるいは全てのファイルについての読み出し履歴情報の読み出しが終了するまで、ステップS63～S65が繰り返される。

【0101】このように、最終の読み出し日時を記録することによって、より最近に参照されたファイルなるべく消去あるいは圧縮しなくすることができるため、効率的な空き領域確保が行える。また、最終の読み出し日時を前述した読み出し回数と組み合わせて使用すれば、読み出し回数が少なくても最終の読み出し日時が新しいものについては消去／圧縮対象から除外することかできるようになり、さらに効率的な空き領域確保が可能になる。

【0102】次に、図21を参照して、図15の読み出し履歴情報（読み出しの有無）を利用したファイル自動消去／圧縮動作の別の例を説明する。ここでは、読み出されたことがあるファイルを優先的に消去あるいは圧縮するようにしたものである。

【0103】まず、実施形態1～3と同じ手法で記憶媒体30のファイル記憶部31に十分な空き領域があるか否か、すなわち消去／圧縮条件が成立しているか否かが調べられる（ステップS71）。消去／圧縮条件が成立したならば、消去／圧縮ファイル選択部204の処理が開始される。すなわち、まず、消去／圧縮ファイル選択部204は、ファイル読み出し履歴記憶部32から1つのファイルに関する読み出し履歴情報を読み出し（ステップS73）、そのファイルが読み出されたことがあるファイルであるか否かを調べる（ステップS74）。読み出されたことがあるファイルであれば、消去／圧縮ファイル選択部204は、そのファイル名をファイル消去／圧縮部205に伝えて消去あるいは圧縮させる（ステップS75）。

【0104】この後、再度、消去／圧縮条件が成立するか否かが調べられ（ステップS71）、消去／圧縮条件

(12)

特開平9-128276

21

が成立しなくなるか、あるいは全てのファイルについての読み出し履歴情報の読み出しが終了するまで、ステップS73～S75が繰り返される。

【0105】このように、ユーザがすでに一度参照したことがあるファイルを優先的に消去または圧縮するというアルゴリズムは、電話回線やネットワーク、あるいは電波などの媒体を通じてニュースや天気予報などの文書ファイルを携帯型端末などの電子情報機器にダウンロードし、ユーザがそれらダウンロードしたファイルを必要に応じて参照するという用途において特に有効であり、記憶容量の少ない電子情報機器の記憶媒体を効率良く利用することが可能となる。

【0106】図22には、この発明の第5の実施形態に係る電子情報機器のシステム構成が示されている。

【0107】このシステムは、概略的には実施形態4のファイルシステム20に対して、ファイル書き込み時にファイル読み出し履歴記憶部32の対応する情報を消去する機能を追加したものである。すなわち、この実施形態5で使用されるファイルシステム20は、ファイル読み出し履歴情報を参照して消去または圧縮の対象となるファイルを選択するという点は実施形態4と同じであるが、そのファイル読み出し履歴情報の信頼性を高めるために、ファイル書き込み部206が追加されている。このファイル書き込み部206を利用したファイル読み出し履歴情報の消去動作を図23に示す。

【0108】図23のフローチャートに示されているように、ファイル書き込み時においては、ファイル書き込み部206は、まず、ファイル記憶部31へファイルを書き込みと共に、ファイル読み出し履歴作成部203に該当ファイルのファイル名を通知する（ステップS81、S82）。ファイル読み出し履歴作成部203は、そのファイル名で指定されるファイルのファイル読み出し履歴情報を消去、あるいは初期値にリセットする（ステップS83）。

【0109】例えば、図24に示されているように、読み出しの有無を示すファイル読み出し履歴情報を使用している場合においては、ファイル名i3のファイルについての書き込みが行われたとき、そのファイル名i3のファイルに対応する読み出し履歴情報が「0」にリセットされ、これによってそのファイルは読み出されたことがないファイルとして扱われる。

【0110】このようにファイル書き込みに応じて該当するファイル読み出し履歴情報を初期化する事により、ユーザがすでに一度参照したデータは消してもよいが、まだ参照していないデータは消してはならない、という用途において、更新後のデータをユーザが読み出したかどうかまでを考慮して消去対象のファイルを選択できるようになる。

【0111】図25には、この発明の第6の実施形態に係る電子情報機器のシステム構成が示されている。

22

【0112】このシステムで使用されるファイルシステム20は、ファイル自動消去またはファイル自動圧縮による空き領域確保機能の実現のために、図示のように、ファイル入出力部301、ファイル書き込み部302、ファイルサイズ情報作成部303、消去／圧縮ファイル選択部304、およびファイル消去／圧縮部305を備えている。

【0113】ファイルの書き込み時には、ファイル書き込み部302は、ファイル入出力部301を通じてファイル記憶部31にファイルを書き込むほか、該当ファイルのファイル名およびファイルサイズをファイルサイズ情報作成部303に通知する。ファイルサイズ情報作成部303は、ファイル名とファイルサイズとの対応を示す表を生成し、それをファイル入出力部301を通じてファイルサイズ情報記憶部33に記憶する。このファイルサイズ情報は、各ファイルのディレクトリエントリの一部に格納する事もできる。ファイル消去または圧縮の際には、消去／圧縮ファイル選択部304が、ファイルサイズ情報記憶部33を参照して、各ファイルのサイズを得る。そして、サイズがある決められた値を超えるファイルを選択し、それらのファイルのファイル名をファイル消去／圧縮部305に伝える。ファイル消去／圧縮部305は消去／圧縮ファイル選択部304から得たファイル名のファイルをファイル記憶部31から消去する。

【0114】次に、図26を参照して、ファイルサイズ情報を利用したファイル自動消去／圧縮動作を説明する。ここでは、サイズの大きなファイルが優先的に消去または圧縮される。

【0115】まず、実施形態1～3と同じ手法で記憶媒体30のファイル記憶部31に十分な空き領域があるか否か、すなわち消去／圧縮条件が成立しているか否かが調べられる（ステップS91）。消去／圧縮条件が成立したならば、消去／圧縮ファイル選択部304の処理が開始される。すなわち、まず、消去／圧縮ファイル選択部304は、ファイルサイズ情報記憶部33から1つのファイルに関するサイズ情報を読み出し（ステップS93）、そのファイルサイズが予め決められた基準サイズを超えているか否かを調べる（ステップS94）。基準サイズよりもファイルサイズが大きいファイルであれば、消去／圧縮ファイル選択部304は、そのファイル名をファイル消去／圧縮部305に伝えて消去または圧縮させる（ステップS95）。

【0116】この後、再度、消去／圧縮条件が成立するか否かが調べられ（ステップS91）、消去／圧縮条件が成立しなくなるか、あるいは全てのファイルについてのサイズ情報の読み出しが終了するまで、ステップS93～S95が繰り返される。

【0117】このように、サイズが大きなファイルを優先的に消去または圧縮することによって、ファイル記憶

(13)

特開平9-128276

23

部31の容量を空けるために消去または圧縮するファイルの個数を抑えることができ、ユーザが読もうとしたファイルがすでに消去されている、あるいは圧縮されておりその読み出しに時間がかかる、という問題の発生確率を低減することができる。

【0118】また、このようにサイズが大きなファイルを優先的に消去または圧縮するという処理は、実施形態4および5で説明したファイル読み出し履歴情報と組み合わせることもできる。この場合には、まず、ファイル読み出し履歴情報を参照することによって複数の候補ファイルを選定し、その中でサイズが大きなファイルを優先的に消去または圧縮するという方法を採用することが好ましい。

【0119】図27には、この発明の第7の実施形態に係る電子情報機器のシステム構成が示されている。

【0120】このシステムで使用されるファイルシステム20は、ファイル自動消去または圧縮による空き領域確保機能の実現のために、図示のように、ファイル入出力部401、ファイル変換部402、ファイル内容対応関係部403、消去／圧縮ファイル選択部404、およびファイル消去／圧縮部405を備えている。

【0121】ファイル変換部402は、ファイル記憶部31からあるファイルf aの内容を読み出し、その内容に変換（ファイルデータの圧縮／伸張、文書ファイル形式の変換など）を加えた上で変換後のファイルをファイル記憶部31中の別のファイルf bとして格納する。この場合、ファイル変換部402は、ファイルf aとf bが互いに変換可能な内容を持ちデータ形式だけが異なるファイルであることを、ファイル内容対応関係作成部403に通知する。

【0122】ファイル内容対応関係作成部403は、ファイル変換部402からの通知に基づき、ファイル記憶部31に格納されているファイルの中で相互に変換可能な内容を持つファイル同士の関係を示すファイル内容対応関係情報を生成し、それをファイル内容対応関係記憶部34に格納する。

【0123】ファイル内容対応関係情報としては、各ファイルのディレクトリエントリに登録されるファイル拡張子などを利用する事もできる。すなわち、例えばあるファイルf aを圧縮する事によって得られた圧縮ファイルf bに対してファイルf aと同じファイル名を付与し、そのファイルタイプを示す拡張子としてデータ圧縮形式を示す情報を付与すれば、ファイルf a、f bそれぞれのファイル名と拡張子の関係から、ファイルf a、f bが相互に変換可能な内容を持つファイルであることを認識できる。したがって、この方式を採用すれば、ファイル内容対応関係記憶部34をディレクトリの一部として実現する事ができる。

【0124】ファイル消去または圧縮の際には、まず、消去／圧縮ファイル選択部404がファイル内容対応関

24

係記憶部34を参照して、他のファイルの内容から復元可能なファイルを選択し、そのファイルのファイル名をファイル消去／圧縮部405に伝える。ファイル消去／圧縮部405は消去／圧縮ファイル選択部404から得たファイル名のファイルをファイル記憶部31から消去または圧縮する。

【0125】次に、図28を参照して、ファイル内容対応関係記憶部34を利用したファイル自動消去／圧縮動作を説明する。ここでは、他のファイルからの変換によって内容が復旧できるファイルが優先的に消去または圧縮される。

【0126】まず、実施形態1～3と同じ手法で記憶媒体30のファイル記憶部31に十分な空き領域があるか否か、すなわち消去／圧縮条件が成立しているか否かが調べられる（ステップS101）。消去／圧縮条件が成立したならば、消去／圧縮ファイル選択部404の処理が開始される。すなわち、まず、消去／圧縮ファイル選択部404は、ファイル内容対応関係記憶部34から1つのファイルに関する情報を読み出し（ステップS103）、そのファイルが他のファイル内容から復元可能なファイルであるか否かを調べる（ステップS104）。復元可能なファイルであれば、消去／圧縮ファイル選択部404は、そのファイル名をファイル消去／圧縮部405に伝えて消去または圧縮させる（ステップS105）。

【0127】この後、再度、消去／圧縮条件が成立するか否かが調べられ（ステップS101）、消去／圧縮条件が成立しなくなるか、あるいは全てのファイルについての対応関係情報の読み出しが終了するまで、ステップS103～S105が繰り返される。

【0128】このように、他のファイルからの変換によって内容が復元できるようなファイルを優先的に消去または圧縮し、復元できないファイルに対しては消去や圧縮の処理を行わないようにする制御は、空き領域確保のための自動ファイル消去を採用した場合に特に有効であり、ユーザが読もうとしたファイルがすでに消去されていて読めない、という問題の発生を減らすことができる。また、このように復旧可能なファイルを優先的に消去するという処理は、実施形態4および5で説明したファイル読み出し履歴情報と組み合わせることもできる。この場合には、まず、ファイル読み出し履歴情報を参照することによって複数の消去対象候補を選定し、その中で復旧可能なファイルを優先的に消去するという方法を採用することが好ましい。

【0129】図29には、この発明の第8の実施形態に係る電子情報機器のシステム構成が示されている。

【0130】このシステムで使用されるファイルシステム20は、ハイパーテキストのように他のファイルとの間で参照関係を持つファイルに好適な空き領域自動確保機能の実現のために、図示のように、ファイル入出力部

(14)

特開平9-128276

25

501、参照関係読み出し部502、消去／圧縮ファイル選択部503、およびファイル消去／圧縮部504を備えている。

【0131】ファイル消去または圧縮の際には、まず、消去／圧縮ファイル選択部503からの指示により参照関係読み出し部502が動作を開始し、ファイル記憶部31の各ファイルについて他のファイルからの参照数が検出される。参照数は、他のファイルそれぞれの内容を解析して、該当ファイルへのポインタの数を調べる事によって検出される。消去／圧縮ファイル選択部503は、他のファイルからの参照数がある一定数よりも少ないファイルを選択し、そのファイルのファイル名をファイル消去／圧縮部504に伝える。ファイル消去／圧縮部504は消去／圧縮ファイル選択部503から得たファイル名のファイルをファイル記憶部31から消去する。

【0132】次に、図30を参照して、ファイル参照数を利用したファイル自動消去／圧縮動作を説明する。ここでは、他のファイルからそのファイルへの参照の個数の少ないファイルが優先的に消去または圧縮される。

【0133】まず、実施形態1～3と同じ手法で記憶媒体30のファイル記憶部31に十分な空き領域があるか否か、すなわち消去／圧縮条件が成立しているか否かが調べられる（ステップS111）。消去／圧縮条件が成立したならば、消去／圧縮ファイル選択部503の処理が開始される。すなわち、まず、消去／圧縮ファイル選択部503は、ファイル記憶部31から1つのファイルを読み出し（ステップS113）、そのファイルについての他のファイルからの参照数を参照関係読み出し部502に調べさせる。参照関係読み出し部502は、他のファイルそれぞれの内容を解析して、該当ファイルへのポインタの数を調べる事によって参照数を検出し、その結果を消去／圧縮ファイル選択部503に渡す。消去／圧縮ファイル選択部503は、そのファイルに対応する参照数が予め定められた基準値よりも少ないか否かを調べる（ステップS114）。他のファイルからの参照数が基準値よりも少ないファイルであれば、消去／圧縮ファイル選択部503は、そのファイル名をファイル消去／圧縮部504に伝えて消去または圧縮させる（ステップS115）。

【0134】この後、再度、消去／圧縮条件が成立するか否かが調べられ（ステップS111）、消去／圧縮条件が成立しなくなるか、あるいは全てのファイルについての他のファイルからの参照数が調べられるまで、ステップS113～S115が繰り返される。

【0135】このように、他のファイルからの参照が少ないファイルを優先的に消去または圧縮し、参照が多いファイルをそのままの形で保存することによって、特に空き領域確保機能のためにファイル消去を採用した場合

26

したファイルがすでに消去されている、という問題の発生を減らすことができる。また、このように参照数の少ないファイルを優先的に消去するという処理は、実施形態4および5で説明したファイル読み出し履歴情報と組み合わせて使用することもできる。この場合には、まず、ファイル読み出し履歴情報を参照することによって複数の消去対象候補を選定し、その中で他のファイルからの参照数の少ないファイルを優先的に消去するという方法を採用することが好ましい。

【0136】なお、実施形態4～8のファイルシステム20のファイル自動消去／圧縮機能は、実施形態1～3のファイルシステム2の場合と同様に、オペレーティングシステムの外でユーザレベルで動くプログラムとして実現することもできる。この場合、図14、図22、図25、図27、図29それぞれのファイル入出力部だけがオペレーティングシステムにファイル管理部として実装され、他の部分はオペレーティングシステム外で動作するファイル管理プログラムとして実装されることになる。

【0137】以上説明したように、実施形態4～8の電子情報機器によれば、ファイルの参照履歴・大きさ・内容の対応関係・参照の個数などが消去または圧縮ファイル選定のための優先度として使用されることにより、静的な優先度だけを使用する場合よりもユーザに与える影響の少ないファイル自動消去を行う事ができる。

【0138】なお、実施形態4～8の電子情報機器においても、実施形態2、3で説明したように、消去／圧縮ファイル選択部により選択されたファイルをユーザに通知し、消去／圧縮の許可を得た上でそのファイルを消去または圧縮するという機能を適宜組み合わせることが好ましい。

【0139】次に、この発明の第9の実施形態を説明する。

【0140】この実施形態9の電子情報機器は、前述した実施形態1～3のシステム構成と同様に、アプリケーションプログラムなどのユーザプログラム100と、そのプログラム100からファイル操作に関する要求を受けてファイルの入出力を管理するファイルシステム20と、ファイルを記憶するための補助記憶装置として使用されるハードディスクやフラッシュメモ리카ードなどの不揮発性の記憶媒体300とから構成される。

【0141】ファイルシステム200は、記録媒体300の空き容量を自動的に確保するための機能として、ファイル自動消去機能とファイル自動圧縮機能の双方を有しており、それらファイル自動消去機能とファイル自動圧縮機能を選択的に使用する構成である。

【0142】ファイルシステム200は、プログラム100からの要求を受理する要求受理部601と、消去すべきファイルを選択する消去ファイル選択部602と、ファイルを消去するファイル消去部603と、圧縮すべ

(15)

特開平9-128276

27

きファイルを選択する圧縮ファイル選択部604と、ファイルを圧縮するファイル圧縮部605と、記憶媒体300に記憶されるファイルを入力するファイル入出力部606と、これらを司る制御部607とからなる。

【0143】図32はファイルシステム200で管理されるファイル構造を示している。

【0144】各々のノードの第1フィールドはノードの種類を表す。Dはディレクトリ、Fはファイルである。第2フィールドはファイルまたはディレクトリの名前である。ディレクトリの場合、第3フィールド以降は子ノードへのポインタである。斜め線は終了を示し、ポインタとして使用されない値を使えば良い。例えばポインタとしてノードの番地を格納し、終了の記しとしてマイナス1を使えば良い。ファイルの場合、第3フィールドに最終更新日時を、第4フィールドに最終参照日時を示す。／は参照されていないことを示す。第5フィールドはデータへのポインタである。

【0145】図33はファイル管理方式のアルゴリズムを示すフローチャートである。本実施形態ではファイルの最終参照日時を圧縮や消去ファイルを選択するための属性として使用する。

【0146】まず、ファイルシステム200の要求受理部601が、ユーザプログラム100からのファイル操作要求を受け付ける（ステップS201）。そのファイル操作要求がファイル書き込み要求でなければ（ステップS202）、そのファイル操作要求によって指定されたファイル操作（ファイル読み出し、ファイル消去、ディレクトリ参照など）が通常通り行われる（ステップS203）。

【0147】一方、ユーザプログラム100からのファイル操作要求が書き込み要求であるならば（ステップS202）、制御部607は記憶媒体300の空き容量に従って、所定の消去／圧縮条件が成立しているか否かを判断する（ステップS204）。この場合、書き込み要求されたファイルサイズと記憶媒体300の現在の空き容量とが比較され、空き領域のサイズが書き込み要求されたファイルサイズよりも小さい時に消去／圧縮条件が成立すると判断される。また、書き込み要求されたファイルサイズとは無関係に、記憶媒体300の現在の空き容量が規定値以下であれば消去／圧縮条件が成立すると判断してもよい。

【0148】十分な空き容量があって消去または圧縮の条件が成立しないならば（ステップS205）、要求されたファイルの書き込みが直ぐに実行される（ステップS206）。一方、十分な空き容量がなく消去／圧縮条件が成立したならば（ステップS205）、制御部607によって、記憶媒体300に圧縮してもよいと判断される圧縮可能ファイルが存在するか否かが判断される（ステップS207）。この判断は、例えば、全てのファイルが圧縮済みか否かを調べるなどによって行う

28

ことができる。この場合には、非圧縮ファイルが残っていれば圧縮可能ファイルが存在すると判断され、残っていなければ圧縮可能ファイルが存在しないと判断される。また、ディレクトリやファイル拡張子などによって予め圧縮してはいけないファイルと圧縮してもよいファイルをユーザによる明示的な指定などによって固定的に決めておき、それに基づいて圧縮可能ファイルが存在するか否かを判断する事もできる。

【0149】記憶媒体300に圧縮してもよいと判断される圧縮可能ファイルが残っているならば、圧縮ファイル選択部604は、圧縮してもよいファイルの中で最終参照日時の最も古いもの（参照されたことのないファイルの最終参照時刻は最も古いと解釈される）を圧縮対象ファイルとして選択し、これがファイル圧縮部605によって圧縮される（ステップS208、S209）。この後、再び消去／圧縮の条件が成立しているかどうかの判断処理が行われ（ステップS204、S205）、成立しなくなるまでは、圧縮してもよいファイルが残っている限り、ステップS208、S209が繰り返し実行される。

【0150】一方、圧縮してもよいファイルが無い場合には、今度は、空き領域確保のために行われる処理が、ファイル圧縮からファイル消去に切り替えられる。

【0151】この場合、消去ファイル選択部602は、全てのファイルの中で最終参照日時の最も古いものを消去対象ファイルとして選択し、これがファイル消去部603によって消去される（ステップS210、S211）。

【0152】この後、再び消去／圧縮の条件が成立しているかどうかの判断処理が行われ（ステップS204、S205）、成立しなくなるまで、ステップS210、S211が繰り返し実行される。

【0153】このように、この実施形態では、ファイル自動圧縮処理がファイル自動消去処理よりも優先して実行するように制御されるので、空き領域の自動確保機能の実行によってユーザに与える影響をより少なく抑えることができる。

【0154】参照されたことのないファイルの最終参照時刻は最も古いと解釈すればよい。これを繰り返せば、書き込むファイルに同等以上の空き領域を作ることができる。

【0155】なお、ここでは、圧縮と消去の対象ファイルを選択する際に、どちらも最終参照時刻という同一の属性を用いたが、必ずしもこの限りではない。圧縮は最終参照時刻の最も古いものを選択し、消去はファイルサイズの大きいものを選択する、というように異なる属性を使用しても良い。

【0156】図34はファイル管理方式の別のアルゴリズムを示すフローチャートである。図33に示す例では、例えば全ての非圧縮ファイルを圧縮してから消去を

(15)

特開平9-128276

29

するため、引き続いて使用される可能性の高いファイルも圧縮されてしまう恐れがある。ファイルの圧縮にはCPU時間や電力を消費すること、また空き領域がない状態にすぐに戻ってしまうことにより、すぐに使われるファイルを圧縮するのは得策ではない。これを改善するためには、すぐに使用される可能性が高いファイルは非圧縮状態であっても圧縮せずに、むしろ使用される可能性の低いファイルを消去した方がよい。一般に使用されるか否かの判定は困難であるため、時間的局所性を仮定し、最終参照日時が新しいものほど使用される可能性が

【0157】まず、ファイルシステム200の要求受理部601が、ユーザプログラム100からのファイル操作要求を受け付ける（ステップS301）。そのファイル操作要求がファイル書き込み要求でなければ（ステップS302）、そのファイル操作要求によって指定されたファイル操作（ファイル読み出し、ファイル消去、ディレクトリ参照など）が通常通り行われる（ステップS303）。

【0158】一方、ユーザプログラム100からのファイル操作要求が書き込み要求であるならば（ステップS302）、制御部607は記憶媒体300の空き容量に従って、所定の消去／圧縮条件が成立しているか否かを判断する（ステップS304）。この場合、書き込み要求されたファイルサイズと記憶媒体300の現在の空き容量とが比較され、空き領域のサイズが書き込み要求されたファイルサイズよりも小さい時に消去／圧縮条件が成立すると判別される。また、書き込み要求されたファイルサイズとは無関係に、記憶媒体300の現在の空き容量が規定値以下であれば消去／圧縮条件が成立すると判別してもよい。

【0159】十分な空き容量があって消去または圧縮の条件が成立しないならば（ステップS305）、要求されたファイルの書き込みが直ぐに実行される（ステップS306）。一方、十分な空き容量がなく消去／圧縮条件が成立したならば（ステップS305）、圧縮してもよいファイルである例えば非圧縮ファイルの中から最も最終参照日時の古いものを選びFcとし、そのFcの最終参照日時をt（Fc）とする（ステップS307）。次に、全ての管理対象ファイルの中から最も最終参照日時の古いものを選びFdとし、そのFdの最終参照日時をt（Fd）とする（ステップS307）。

【0160】次に、現在の日時をtとすると、 $|t(Fd) - t|$ と $|t(Fc) - t|$ の比があらかじめ定めた定数pを超えるか否か、つまり $|t(Fd) - t| / |t(Fc) - t| > p$ が成立するか否かが調べられる（ステップS309）、成立しているとき、つまり圧縮対象ファイルFcの最終参照日時よりもFdが著しく古いファイルであれば、F

30

dの消去が行われる（ステップS311）。一方、ステップS309の条件が成立していないときは、圧縮対象ファイルFcの圧縮が行われる（ステップS310）。

【0161】最終変更日時を圧縮や消去の属性として使用する場合には、圧縮によりできたファイルの最終変更日時は非圧縮状態の最終変更日時を継承するものとする。また、最終参照日時を圧縮や消去の属性として使用する場合には、圧縮や消去の対象とするために行ったファイルの属性へのアクセスは参照とはみなさず最終参照日時に変更されないものとする。

【0162】なお、以上の実施形態1～9では、ファイル管理の対象を特に記述しなかったが、管理の対象は限定されても良い。例えばあるドライブのファイルシステムだけが当該ファイルの管理対象になる、あるドライブ内の特定のファイルシステムだけが当該ファイルの管理対象になる、あるディレクトリの下だけが当該ファイルの管理対象になる、あるアプリケーションが使用するファイルだけが当該ファイルの管理対象になる、などの管理方法を使用して圧縮または削除対象となりうるファイルを制限してもよい。また、前述したようにファイル管理は必ずしもOSの一部である必要はない。別のモジュールとして実現されても良い。あるいは、あるアプリケーションの使用するファイルだけが当該ファイルの管理対象になる場合はそのアプリケーションが当該ファイル管理を行っても良い。

【0163】また、実施形態1～9のファイル管理方法はコンピュータによって実行可能なプログラムとして実現されているため、フロッピーディスクやCD-ROMなどの記録媒体にそのプログラムを格納して配布することにより、それら媒体からそのプログラムをコンピュータにインストールするだけで、簡単にディスクなどの2次記憶装置の空き領域を管理することが可能となる。

【0164】

【発明の効果】以上説明したように、この発明によれば、ファイルシステムが管理する記憶媒体の空き領域に応じてファイル消去あるいは圧縮による空き領域確保処理が自動的に行われるため、ユーザが消去すべきファイルを選択したり、消去するためのプログラムの使用法を覚える必要がなく、使い勝手の良い計算機システムを提供することができる。また、外部からデータが転送されてくる場合に、ユーザが空き領域を作らなくても転送データを失わずファイルシステムに格納することができる。さらに、ファイルの参照履歴、内容の対応関係、参照の個数を利用して消去を行うことにより、ユーザが参照する可能性が低いファイルを消去できるようになり、自動消去の利便性を向上することができる。

【図面の簡単な説明】

【図1】この発明の第1実施形態に係る計算機システムで使用されるファイルシステムの機能構成を示すブロック図。

(17)

特開平 9-128276

31

【図 2】同第 1 実施形態のシステムで扱われるファイルの階層構造を示す図。

【図 3】同第 1 実施形態のシステムで扱われるファイルの内部構成を示す図。

【図 4】同第 1 実施形態のシステムにおけるファイル自動消去処理の手順を示すフローチャート。

【図 5】同第 1 実施形態のシステムに対して書き込み要求を行うプログラムの一例を示す図。

【図 6】同第 1 実施形態のシステムにおけるファイル自動消去処理で用いられる消去条件判定処理を説明するための図。

【図 7】同第 1 実施形態のシステムにおけるファイル自動消去処理で実行される消去条件判定処理および消去ファイル選択処理の手順を示すフローチャート。

【図 8】この発明の第 2 実施形態に係る計算機システムで使用されるファイルシステムの機能構成を示すブロック図。

【図 9】この発明の第 3 実施形態に係る計算機システムで使用されるファイルシステムの機能構成を示すブロック図。

【図 10】同第 3 実施形態のシステムにおけるファイル自動消去処理の手順を示すフローチャート。

【図 11】同第 3 実施形態のシステムにおいて消去対象ファイルを選択するための参照されるファイルの重要度の一例を示す図。

【図 12】同第 1 実施形態のファイルシステムをオペレーティングシステムと独立したユーザプログラムとして実現した様子を示す図。

【図 13】同第 1 実施形態のファイルシステムをユーザプログラムとして実現した場合におけるそのファイルシステムの機能構成を示すブロック図。

【図 14】この発明の第 4 実施形態に係る計算機システムで使用されるファイルシステムの機能構成を示すブロック図。

【図 15】同第 4 実施形態のシステムで管理されるファイル読み出し履歴情報の一例を示す図。

【図 16】同第 4 実施形態のシステムで管理されるファイル読み出し履歴情報の他の例を示す図。

【図 17】同第 4 実施形態のシステムで管理されるファイル読み出し履歴情報のさらに他の例を示す図。

【図 18】同第 4 実施形態のシステムにおけるファイル自動消去処理の手順を示すフローチャート。

【図 19】同第 4 実施形態のシステムで管理されるファイル読み出し履歴情報のさらに別の例を示す図。

【図 20】同第 4 実施形態のシステムにおける他のファイル自動消去処理の手順を示すフローチャート。

【図 21】同第 4 実施形態のシステムにおけるさらに他のファイル自動消去処理の手順を示すフローチャート。

【図 22】この発明の第 5 実施形態に係る計算機システムで使用されるファイルシステムの機能構成を示すブ

32

ック図。

【図 23】同第 5 実施形態のシステムにおけるファイル読み出し履歴情報の消去処理の手順を示すフローチャート。

【図 24】同第 5 実施形態のシステムにおいてファイル読み出し履歴情報が消去される様子を示す図。

【図 25】この発明の第 6 実施形態に係る計算機システムで使用されるファイルシステムの機能構成を示すブロック図。

【図 26】同第 6 実施形態のシステムにおけるファイル自動消去処理の手順を示すフローチャート。

【図 27】この発明の第 7 実施形態に係る計算機システムで使用されるファイルシステムの機能構成を示すブロック図。

【図 28】同第 7 実施形態のシステムにおけるファイル自動消去処理の手順を示すフローチャート。

【図 29】この発明の第 8 実施形態に係る計算機システムで使用されるファイルシステムの機能構成を示すブロック図。

【図 30】同第 8 実施形態のシステムにおけるファイル自動消去処理の手順を示すフローチャート。

【図 31】この発明の第 9 実施形態に係る計算機システムで使用されるファイルシステムの機能構成を示すブロック図。

【図 32】同第 9 実施形態のシステムで使用されるファイル管理構造の一例を示す図。

【図 33】同第 9 実施形態のシステムにおいてファイル自動消去処理とファイル自動圧縮処理とを選択的に使用した空き領域確保処理の手順を示すフローチャート。

【図 34】同第 9 実施形態のシステムにおいてファイル自動消去処理とファイル自動圧縮処理とを選択的に使用した空き領域確保処理の別の手順を示すフローチャート。

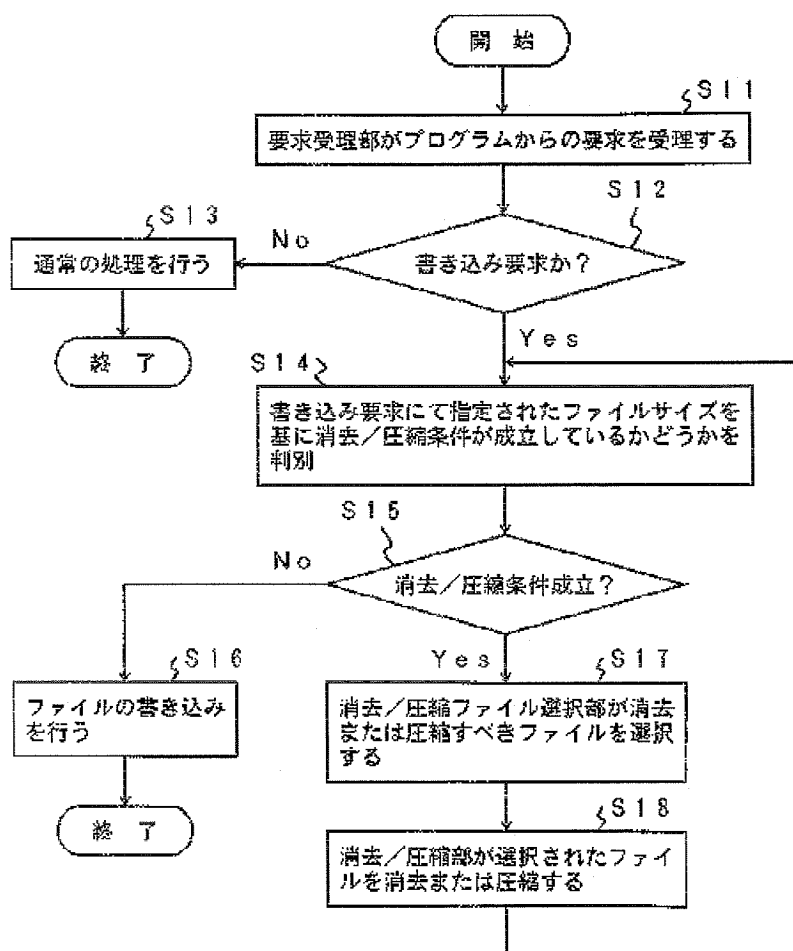
【符号の説明】

1、10…ユーザプログラム、2、20…ファイルシステム、3、30…記憶媒体、4…許可通知プログラム、5…オペレーティングシステム、6…ファイル管理プログラム、21…要求受理部、22…消去/圧縮条件判別部、23…消去/圧縮ファイル選択部、24…ファイル消去/圧縮部、25…ファイル入出力部、26…消去/圧縮通知部、27…諾否受理部、32…ファイル読み出し履歴記憶部、33…ファイルサイズ情報記憶部、34…ファイル内容対応関係記憶部、201…ファイル入出力部、202…ファイル読み出し部、203…ファイル読み出し履歴作成部、204…消去/圧縮ファイル選択部、205…ファイル消去部、206…ファイル書き込み部、303…ファイルサイズ情報作成部、403…ファイル内容対応関係作成部、502…参照関係読み出し部。

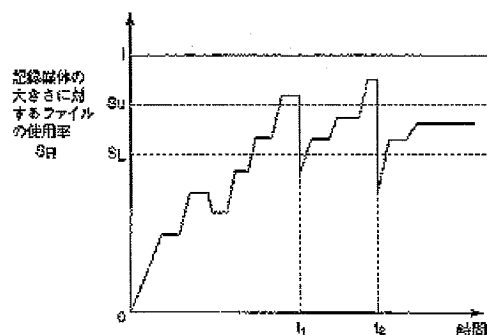
(19)

特開平9-128276

【図4】



【図6】



【図15】

ディレクトリ

ファイル名	ファイルタイプ	読み出し履歴 (読み出しの有無)
f1			0
f2			1
f3			1
f4			0
...

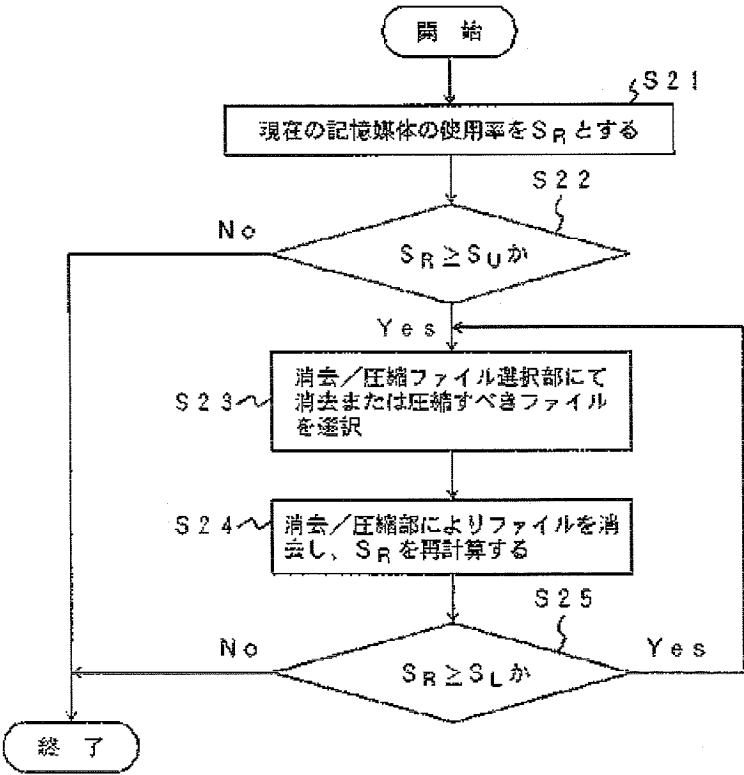
0: 読み出されたことのないファイル

1: 読み出されたことのあるファイル

(20)

特開平9-128276

【図7】



【図24】

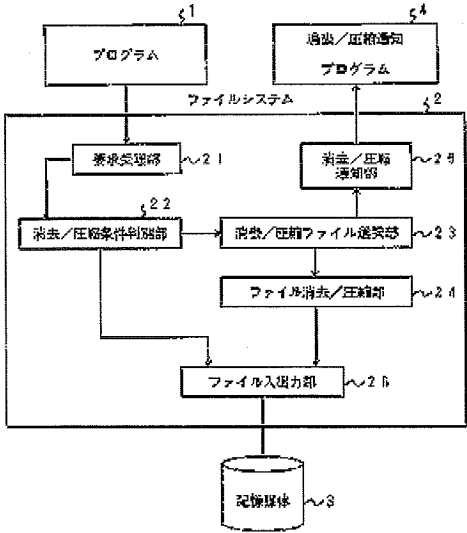
13の読み出し履歴を出力

ファイル名	読み出し履歴 (読み出しの有無)
f1	0
f2	1
f3	1→0
f4	0
...	...

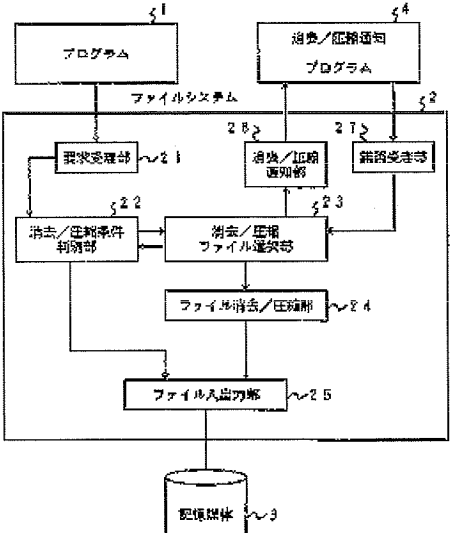
↑3ライト

↑32

【図8】



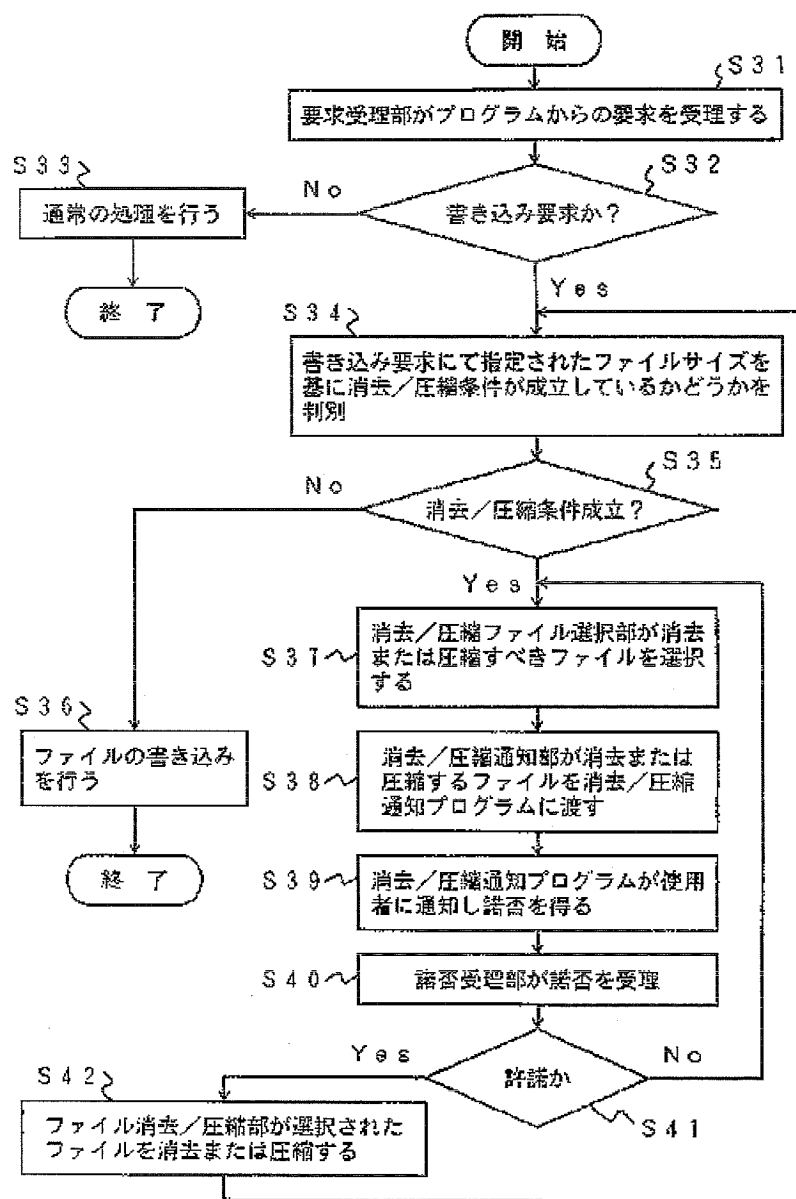
【図9】



(21)

特開平9-128276

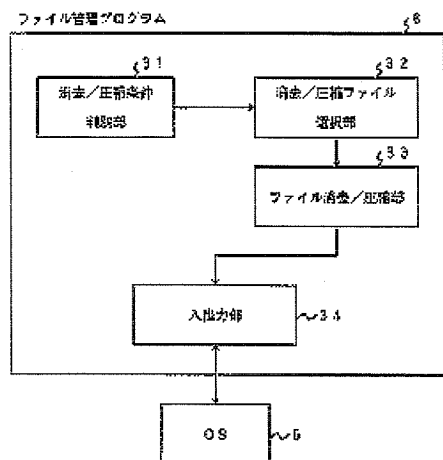
【図10】



(22)

特開平9-128276

【圖 13】



【图 17】

ファイル名	ファイルタイプ	*****

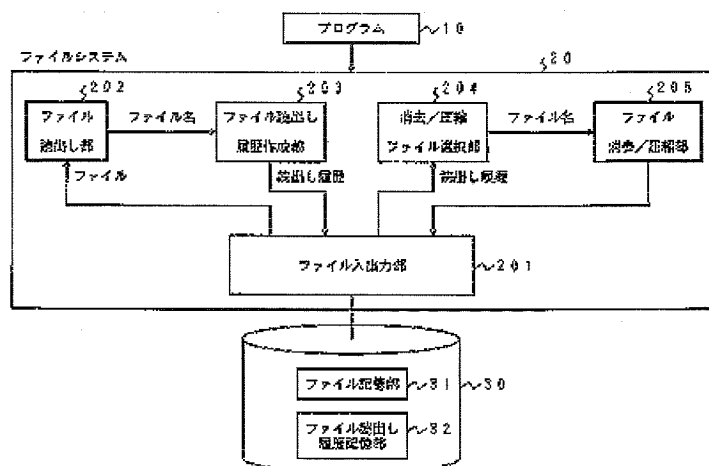
(A)

読出し履歴ファイル

ファイル名	読出し履歴

(10)

【图 14】



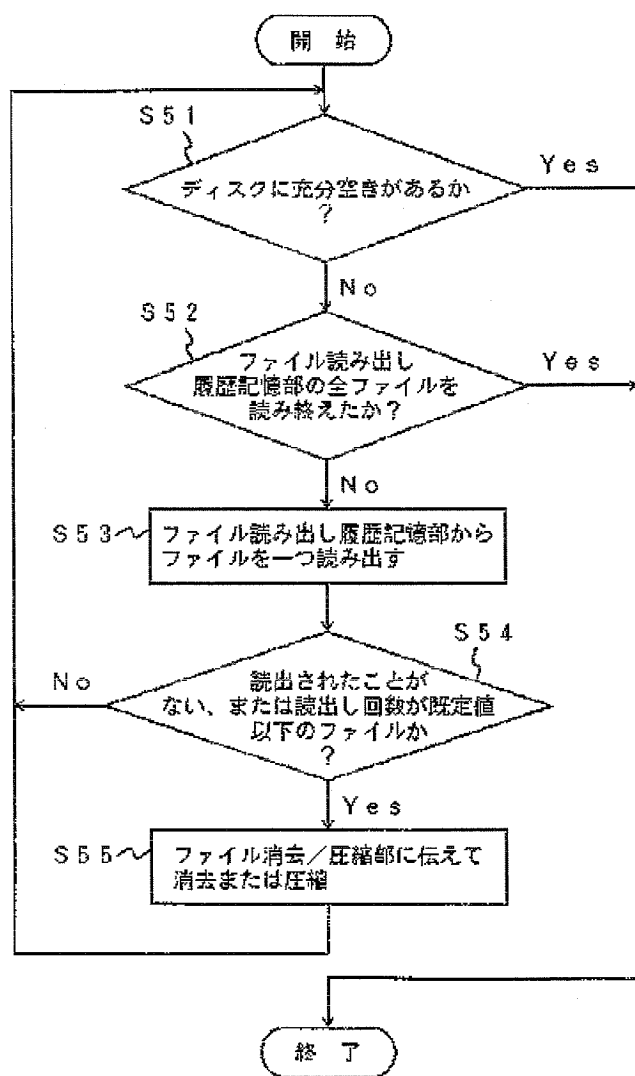
【例 19】

流出し履歴ファイル	
ファイル名	流出し履歴
	(最終読出し時刻)
f1	10:30 95-8-7
f2	17:30 95-8-8
f3	10:30 95-8-6
f4	17:30 95-8-5
f5	10:30 95-8-5

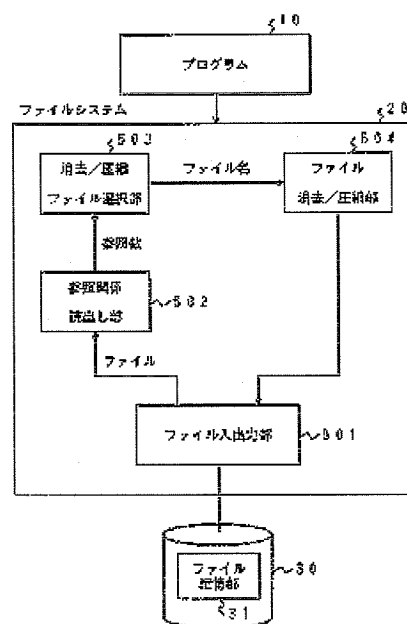
(23)

特開平9-128276

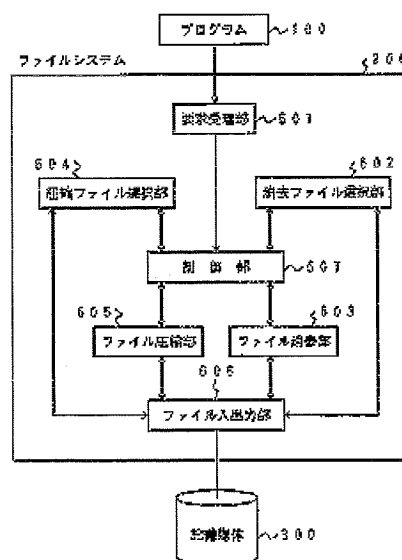
【図18】



【図29】



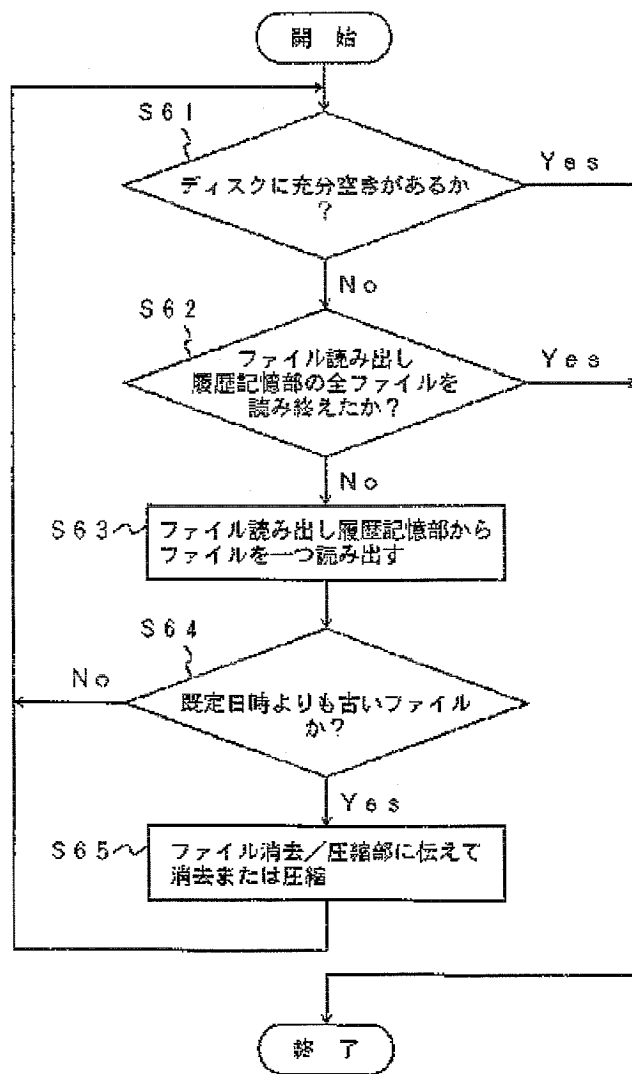
【図31】



(24)

特開平9-128276

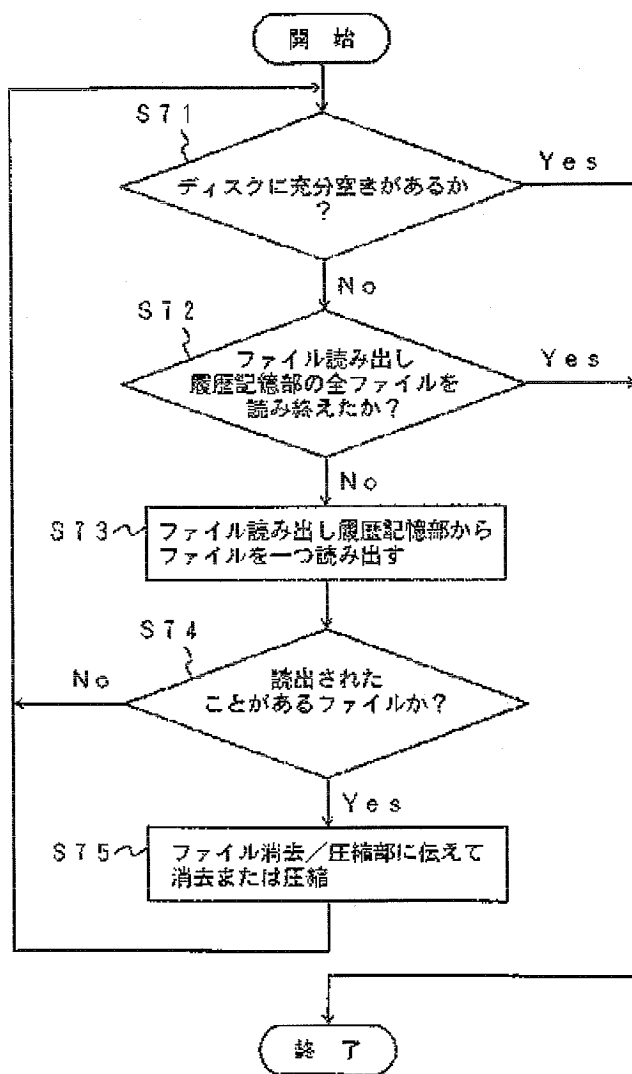
【図20】



(25)

特開平9-128276

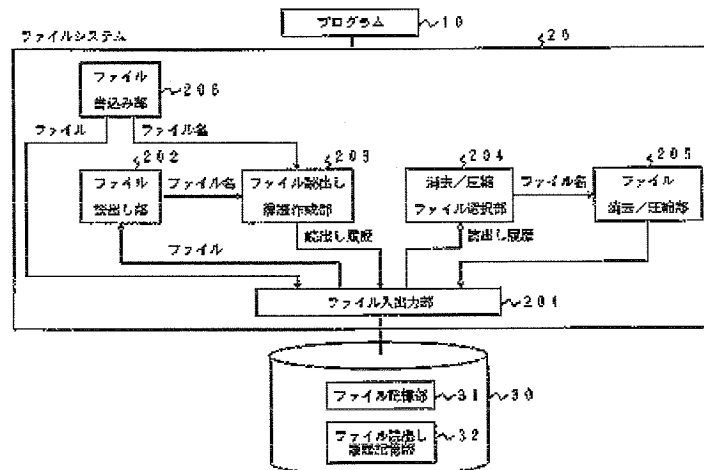
【図21】



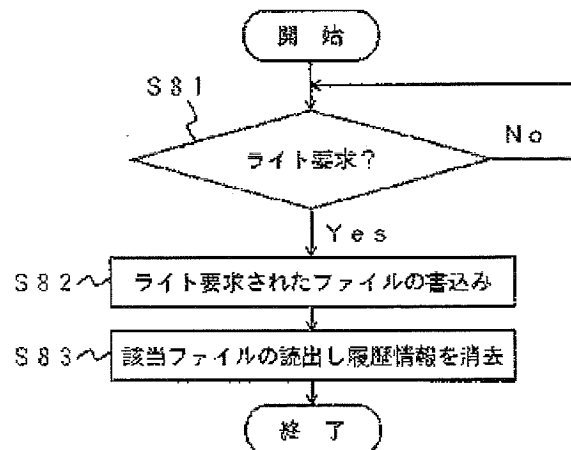
(25)

特開平9-128276

【图22】



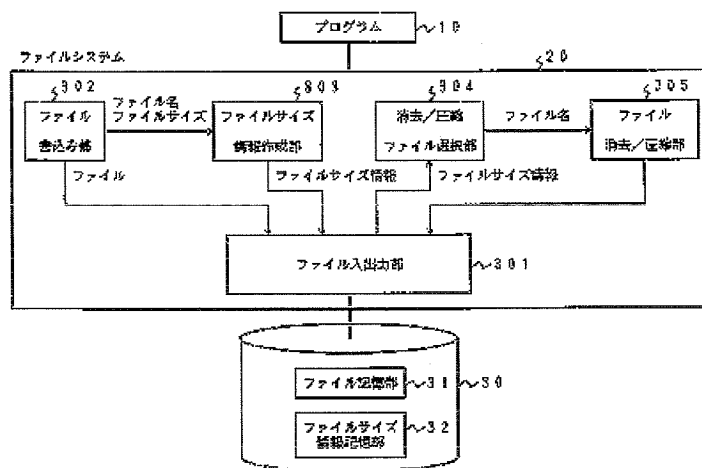
【图23】



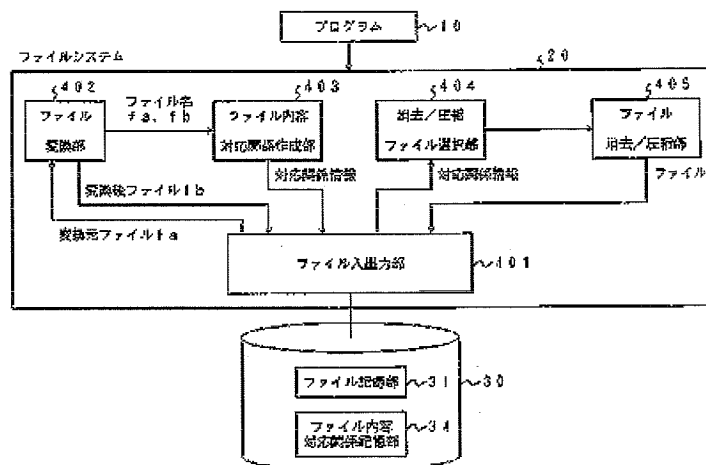
(27)

特開平9-128276

【図25】



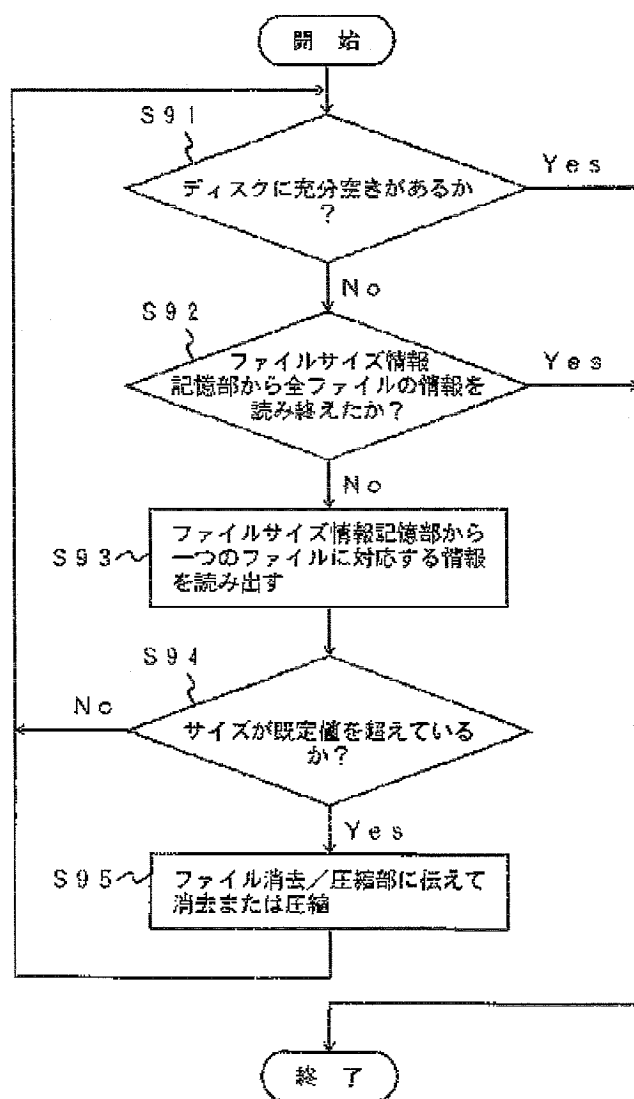
【図27】



(28)

特開平9-128276

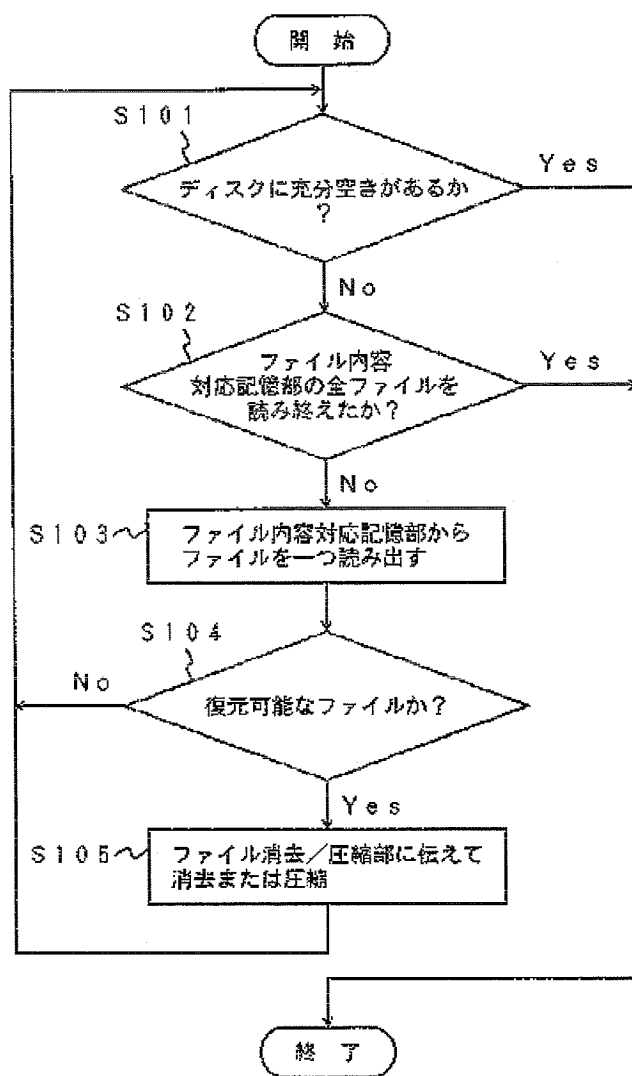
【図26】



(29)

特開平9-128276

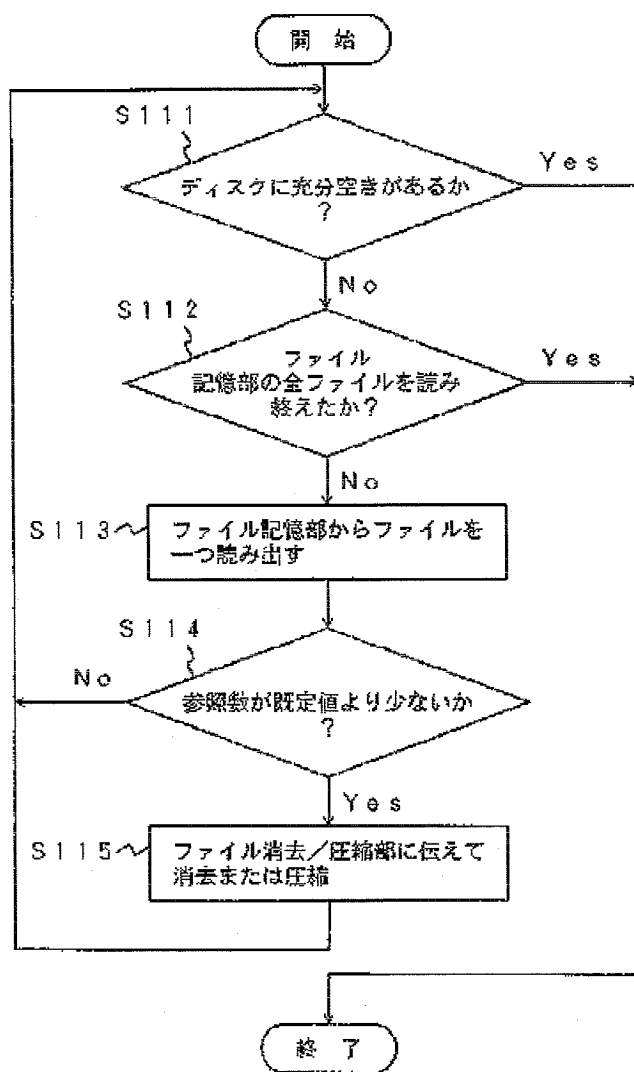
【図28】



(30)

特開平9-128276

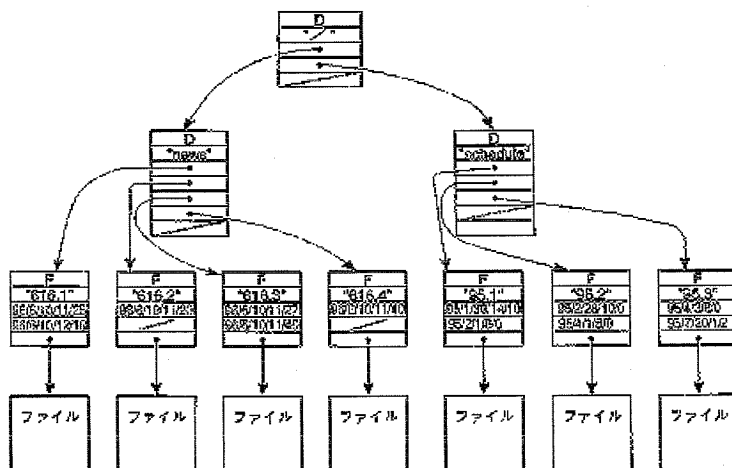
【図30】



(31)

特開平9-128276

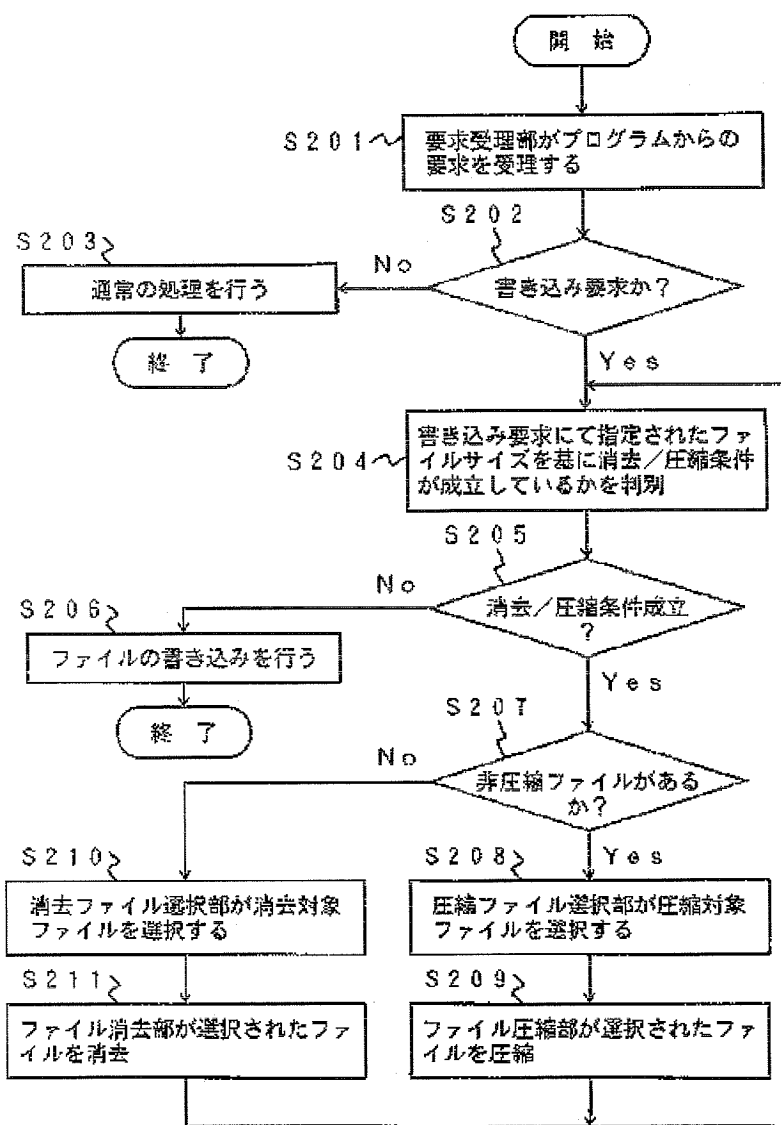
【図32】



(32)

特開平9-128276

【図33】



(33)

特開平9-128276

【図34】

